UNIVERSITY OF OULU

FACULTY OF INFORMATION TECHNOLOGY AND ELECTRICAL ENGINEERING

Ilkka Karjalainen

DIPPAOUT: COMPARING THE PLAYER EXPERIENCE OF DIFFERENT INTERACTION METHODS IN MOBILE GAMES

Master's Thesis
Degree Programme in Computer Science and Engineering
May 2021

Karjalainen I. (2021) DippaOut: Comparing the Player Experience of Different Interaction Methods in Mobile Games. University of Oulu, Degree Programme in Computer Science and Engineering, 85 p.

ABSTRACT

Mobile phones are one of the most common platforms for gaming. They also provide a wide selection of possible interaction modes, such as touch, tilt, and eye-tracking. Different kinds of games can be created with these interaction methods, but game designers constantly develop new ways to play with various accessories and game designs. Since people carry mobile phones everywhere, mobile phones have enabled playing games everywhere; in buses, during school breaks, and in restrooms. The most popular mobile phone platforms are currently Apple's iOS and Google's Android, which provide touching the screen via finger or stylus as a primary interaction method. The game design challenge provides a good user experience since there are no hardware buttons available as in gaming console controllers. The problem that mobile phones lack hardware buttons has forced game developers to think differently and develop games with only a few interactions available.

This thesis focuses on taking a different approach to that problem. What if the user can control the game by rotating the mobile phone or using eye movement. The goal was to create a simple game with five different interaction methods: touching the screen, rotating the mobile phone, closing left and right eyes, using head and eye-tracking. Data was automatically collected with those five interaction methods during gameplay; additional material was collected after each level about gameplay and after test-session with questionnaires about testers' demographics and opinions. The questionnaires were later analyzed to see how those interaction methods differ from each other; the users' views of those methods and the data collected during gameplay are in line with testers' answers.

The gameplay data was analyzed and the most suitable interaction for playing DippaOut was rotating the phone, and the worst was eye tracking. The order of the remaining three interactions was: touch, head tracking, and eye closing. From the end level questionnaire, players' opinions revealed that they did not like playing DippaOut with eye-tracking, and rotating the mobile phone was the most suitable way to play on players' opinions. In players' view, the touch was the second-best interaction method, and the eye closing, which was a new way to play, was ranked in results to be the third-best interaction method. Head Tracking, which some players were familiar with, was in the fourth position. After the testing session, players filled out a postquestionnaire where they could express their opinions and results were quite similar than on other measurements and end level questionnaire.

Keywords: Mobile games, motion control, eye control, custom interaction methods, player experience, game user research, user study.

Karjalainen I. (2021) Ilkka Karjalainen Thesis. Oulun yliopisto, Tietotekniikan tutkinto-ohjelma, 85 s.

TIIVISTELMÄ

Matkapuhelimet ovat yksi yleisimmistä pelialustoista. Ne tarjoavat myös laajan valikoiman mahdollisia vuorovaikutustiloja, kuten kosketus, kallistus ja katseen seuranta. Näillä vuorovaikutustavoilla voidaan luoda erilaisia pelejä, mutta pelisuunnittelijat kehittävät jatkuvasti uusia tapoja pelata erilaisilla lisävarusteilla ja pelien suunnittelulla. Ihmiset kantavat matkapuhelimia kaikkialla ja matkapuhelimet ovat mahdollistaneet pelien pelaamisen näin myös kaikkialla; busseissa, koulumatkojen aikana ja vessoissa. Suosituimmat matkapuhelinalustat ovat tällä hetkellä Applen iOS ja Googlen Android, jotka tarjoavat ensisijaisena vuorovaikutustapana kosketuksen näyttöön sormella tai stylus kynällä. Pelien suunnittelussa on haasteena hyvän käyttökokemuksen tarjoaminen, koska fyysisiä näppäimiä ei ole saatavana kuten pelikonsolien ohjaimissa. Fyysisten näppäimien puuttuminen on ongelma verrattaessa perinteisiin peleihin, mikä on pakottanut pelikehittäjät ajattelemaan erilailla ja kehittämään pelejä eri vuorovaikutuksilla.

Tässä opinnäytetyössä keskitytään mobiilipelien vuorovaikutusongelmaan. Entä jos käyttäjä voi hallita peliä kääntämällä matkapuhelinta tai käyttämällä silmän liikkeitä? Tavoitteena oli luoda yksinkertainen peli viidellä eri vuorovaikutustavalla: näyttöä koskettamalla, matkapuhelinta kääntelemällä, sulkemalla vasenta tai oikeaan silmää, päätä kääntämällä ja katseen seurannalla. Tiedot kerättiin automaattisesti näiden viiden eri vuorovaikutustavan avulla pelin aikana. Pelin jälkeen kerättiin lisämateriaalia tason jälkeisellä kyselylomakkeella ja testisession jälkeisellä kyselomakkeella pelaajien mielipiteistä ja väestötiedoista. Kyselylomakkeita analysoitiin myöhemmin sen selvittämiseksi, miten nämä vuorovaikutustavat eroavat toisistaan. Sekä miten pelaajien näkemykset näistä menetelmistä ja pelin aikana kerätyt tiedot eroavat toisistaan.

Pelitiedot analysoitiin ja pelaajakokemukseltaan sopivin vuorovaikutus DippaOut pelin pelaamiseen oli puhelimen kallistaminen ja huonoin oli katseen seuranta. Kolmen muun vuorovaikutuksen järjestys oli: kosketus, pään kääntäminen ja silmien sulkeminen. Pelaajien mielipiteet paljastivat tason jälkeisestä kyselylomakkeesta, että he eivät pitäneet DippaOutin pelaamisesta katseen seurannalla ja matkapuhelimen kallistaminen oli sopivin tapa pelata. Kosketus oli toiseksi paras vuorovaikutusmenetelmä, ja silmien sulkeminen, joka oli uusi tapa pelata, luokiteltiin tuloksissa kolmanneksi parhaaksi vuorovaikutustavaksi. Joillekin pelaajille tuttu pään kääntäminen oli neljänneksi paras vuorovaikutusmenetelmä.

Avainsanat: Mobiilipelit, liikkeen hallinta, silmien hallinta, mukautetut vuorovaikutustavat, pelaajakokemus, pelien käyttäjätutkimus, käyttäjätutkimus.

TABLE OF CONTENTS

FOREWORD

This thesis finalizes my university studies which have lasted more than a decade. The subject was selected due to my interest in mobile games and outside-box thinking and refined to research with Paula Alavesa. She is the supervisor of this thesis. I want to thank Paula Alavesa for pushing me through this thesis and my family for supporting my University studies, and Markku Suomalainen, who also supervised this thesis, guiding me to focus on the right things from the noise. Also, my children were a great help for giving ideas and suggestions for the implemented game and acting as game testers. Colleagues from work at Finlabs provided their valuable time for testing and filling out the questionnaire, so big thanks also for their help.

An interesting issue was raised during gameplay sessions where players were developers who were young and studied in Oulu in the 2000ish. Nokia hired students from the University of Oulu, and not all of those managed to graduate, and studies left unfinished when unexpected events in life happened. When I was doing my thesis, many of my colleagues started thinking about finishing their studies, but they would need a little extra push to start.

Oulu, May 13th, 2021

Ilkka Karjalainen

LIST OF ABBREVIATIONS AND SYMBOLS

ACAT	Assistive Context-Aware Toolkit
ADHD	Attention Deficit Hyperactivity Disorder
ALS	Amyotrophic Lateral Sclerosis
ANOVA	Analysis of variance
API	Application programming interface
AR	Augmented reality
ATM	Automated Teller Machines
BC	Before Christ
BCI	Brain-Computer Interface
CD	Compact disc
CISS	Convergence Insufficiency Symptom Survey
EEG	Electroencephalography
FPS	First person shooter
CW	Clockwise
CCW	Counterclockwise
D-pad	Directional pad
IF	Infrared
LCD	Liquid Crystal Display
MMORPG	multiplayer online role-playing game
PDA	Personal digital assistant
ROM	Read-only memory
SD	Standard deviation
SDK	Software Development Kit
SMS	short message service
SQL	Structured Query Language
TTL	transistor–transistor logic
UI	User Interface
VR	Virtual Reality
XML	Extensible Markup Language

1. INTRODUCTION

People have been gaming with different kinds of board games since 3500 before Christ (BC) when a game named Senet was played in Egypt [1]. Different types of games have been created throughout history worldwide. Games have also evolved during the third technology revolution, the digital revolution. Technology has made it possible to make more advanced games, and game prices have also been reduced due to mass production, which has made games available for the broader public. Before electronic games, mechanical games like Pinball and slot machines were created by companies for business purposes. Since people have always liked to play, it is only natural that hobbyists began to experiment with electronic games as soon as possible. The electronic components were expensive, and the information was scarce, so the early electronic games were created in universities by computer scientists [2]. When low-cost programmable microprocessors from MOS Technologies were available in the mid-1970s, they replaced transistor-transistor logic (TTL) and made possible early home consoles and handheld electronic games [3]. In 1973 the first mobile phone call was made by Martin Cooper, a researcher in Motorola [4]. With that phone call, the mobile phone era had begun, and mobile phone evolution made possible the modern, touch-screen, Internet, and video-capable smartphones.

When technology and software made it possible, games were added to mobile phones, first as gimmicks and later as a selling point. In the early mobile phones, the only interaction mode to control games was the keypad, and so the games were designed to be played with that keypad users were familiar with. During that time, games were played on home consoles with joysticks and computers with keyboards and mice. The mobile phone keypad provided the same directions as joysticks; some well-known games like Tetris [5] were ported to mobile phones, and users could easily similarly play the games as with joysticks. When more technology was integrated inside mobile phones, like accelerometers, which are used to detect the orientation of the phone, a new kind of game control could be designed.

Game designers and researchers have embraced new technologies. When more interaction methods have become available in mobile phones and games with different interaction methods have been created as in studies [6, 7, 8]. In those studies, tilt, touch, and facial tracking are compared to each other by collecting data of gameplay with different interactions. The studies also show that the game to be tested needs to be designed to be suitable for multiple interactions, but facial tracking seems to be a substitute for mouse controlling [7, 8, 9].

1.1. Research Method and Question

This thesis explores what kind of different interactions can be used to control mobile games, and what are the differences in player experience between those interactions? This thesis tries to shed light on these questions with a constructive approach and a user study using a research game developed for this purpose [10]. The game is designed to be controlled with five different interaction methods, and facial gestures are not used just mouse substitutes as has been done in previous studies [7, 8, 9]. Because interaction with facial gestures could also be straining, gaming fatigue is examined: does the interaction cause the kind of fatigue that playing is not enjoyable anymore. The research question is: How do different control methods influence player experience in mobile games?

1.2. Authors Contribution

The game created for this thesis is *DippaOut* with a simple controlling interface. In DippaOut, the protagonist is a circle which is constructed with different colored and different sized arcs. The goal is to keep the yellow ball inside the circle, and when the ball hits an arc, the arc will disappear. When all the arcs are hit, and only the ball is left in the in-game area, the level is completed successfully. If the ball escapes the circle, the level is failed. There is, on the game user interface (UI), also a white triangle, which shows the circle's current rotation to help the player rotate the circle in the right direction. The player only needs to rotate the circle either clockwise or counterclockwise.

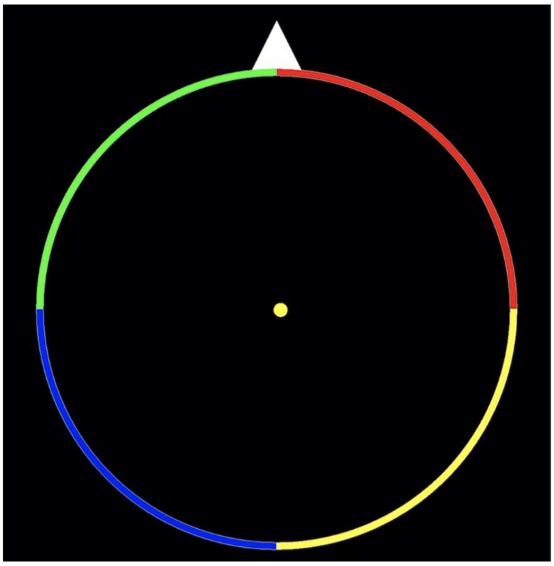

Figure 1. Figure © Ilkka Karjalainen, screenshot from DippaOut showing four different color, same sized arcs

There are three different levels in DippaOut, each more challenging than the previous one. Each level is played with five different interaction methods: Touch, Rotate, Eye Closing, Head Tracking, and Eye Tracking. In this thesis, some data was collected automatically during gameplay. These statistics, i.e., game analytics from gameplay was then analyzed. In addition, the players were presented with a questionnaire to get their opinion and experiences about the gameplay when finishing the level. These opinions were analyzed and compared to gameplay statistics. Also, after the test session, players were given a postquestionnaire to when the player could express opinions about the gameplay, and demographic information was collected.

1.3. Structure of the Thesis

This thesis has the following structure. Chapter 1 is the introduction to this thesis. Chapter 2 opens with the history of digital gaming and mobile games, concludes with the current status, looks at recent research, presents information about different controlling methods, and explains gaming fatigue. Chapter 3 describes the case studies of those controlling methods and presents the implemented testing game. Chapter 4 presents the plan for this study and shows the testing process, and Chapter 5 presents the results. Chapter 6 contains the discussion of the results, and Chapter 7 summarizes the whole thesis. In addition, Chapter 8 presents the references, and Chapter 9 contains the appendices from the data collected from gameplay and questionnaires.

2. RELATED WORK

The idea of playing games with electronic devices dates before the dawn of the first computer. There is a patent for "Cathode-ray tube amusement device" from 1947 [11], which defines a game played with TV. The prototypes were handmade, but the device did not make it into production. The idea of a modern computer, where a machine computes by executing a set of predefined rules (instructions) stored on tape as a program and is therefore programmable, was first introduced by Alan Turing in his seminal paper "On Computable Numbers" [12]. Within the evolution of computers, programming languages also evolved and so did digital games. From the early 1950s, computer scientists developed games for academic research, and for fun [13]. Along with those academic research games, companies and individuals created gaming devices for technology demonstrations like Bertie the Brain [14] and Nimrod [15].

The title of the first electronic game available for the general public and not for academic purposes nor technology demonstration, might be Tennis for Two by William Higinbotham and Robert Dvorak in 1958 [16]. There is always a possibility that someone will discover older games from archives later, but the current consensus is that Tennis for Two holds the title. The game itself is two-player tennis with an oscilloscope as a display. After Tennis for Two, the next big thing was Spacewar! in 1962 [17], which still was using such expensive hardware that people could play it in universities around the USA. In the late 1960s, the electro-mechanical coin-operated game called Periscope was created by Sega [18].

The earliest known coin-operated video game is the Galaxy Game, which is an expanded version of the video game Spacewar. Students created it at Stanford University in 1971 [19]. In the 1970s gaming industry was boosted by technical evolution in microchips. That lead to arcade classics like 1972 founded Atari's Pong [20] and Breakout (with dial control) [21], Taito's Space Invaders (joysticks and buttons) [22], and Sea Wolf from Midway (custom periscope with handles and buttons) [23]. With coin-operated games, the first home console saw daylight in the 1970s. The console was called Magnavox Odyssey and was developed by Ralph H. Baer at Sanders Associates and released by Magnavox. It was still far away from the current consoles providing no sound, no color, and displaying three square dots on the screen. Different kinds of overlays are connected onto the screen to give colors and graphics.

Figure 2. Magnavox Odyssey, first commercial home video game console[24] (Figure: Public domain)

2.1. 1st Generation of Home Consoles

Different companies created the first generation of home consoles in the 70s. A couple of different versions of Magnavox Odyssey were released because the original version was not succeeding due to of hardware limitations. Atari made home versions of arcade hit Pong, Coleco million-selling Telstar consoles [25], and Nintendo's first console, the Color TV-Game 6. There were also dozens of clone consoles and consoles branded for different companies like the retailer Sears. The first generation of home consoles was replaced in the mid-1970s by consoles like Fairchild Channel F, which provided the possibility to change the game by offering games as cartridges plugged directly into the console's CPU. Consoles with changeable games led to the first crash of the console market because customers left the home consoles with fixed games on the shelves.

2.2. 2nd Generation of Home Consoles

The era of these cartridge-based programmable home consoles was called the era of second generation home consoles. With programmable cartridges, companies focusing only on creating games for home consoles were born. The first third-party game developer company was Activision, which made classic games like Pitfall! [26] and continues developing and publishing games still today. Where Fairchild started the second home console generation, Atari followed quickly with Atari 2600, Magnavox with Odyssey^2, Coleco with ColecoVision, and Mattel with Intellivision. With cartridges, players could play conversion of arcade hits like Donkey Kong [27], Space Invaders [22], and Defender [28] on their home consoles. The era came to an end in the video game crash of 1983 (Atari shock). The home console market was saturated with different consoles and the game quality was poor. Third-party game companies copied hit games, played Quick Profit [29], reverse engineered proprietary systems to make games for them and hired programmers from other companies to gain knowledge from proprietary systems. This kind of situation led to lawsuits and feuds between companies. Also, in the 80's the home consoles began facing competition from home computers. The most iconic effect of the 1983 video game crash was Atari, which created poor games like E.T [30], an conversion of Pac-Man [31], discreetly buried about 700.000 cartridges in a landfill (part of those were excavated in 2014 when making the documentary Atari: Game Over).

2.3. 3rd Generation of Home Consoles

The third generation of home consoles was the 8-bit era. Consoles like Nintendo's Family Computer (Famicom) in Japan, Nintendo Entertainment System (NES) the US) and Sega's SG-1000, and Sega Mark III (Sega Master System (SMS)). Home computer manufacturers, e.g., Commodore with C64GS and Amstrad with GX4000, were forced to decline due to the success of Nintendo and Sega. Several companies created many iconic gaming franchises like Super Mario [32], The Legend of Zelda [33], Final Fantasy [34], and Metal Gear [35] during this era. Discontinuation of NES in 1995 was the end of this era. During this era mobile gaming was dominated by

Nintendo with Game Boy but they were no match for console and computer games of that time.

2.4. 4th Generation of Home Consoles

The fourth generation of home consoles was the 16-bit era. The first 16-bit console in the market was TurboGrafx-16 (as PC Engine in Europe and Japan) which was successful in Japan but, due to limited library of games, was not so successful in North America and Europe. The main rivalry was once again between Sega's MegaDrive/Genesis and Nintendo's Super Nintendo Entertainment System (SNES). During this era, companies introduced compact disc read-only memory (CD-ROM) devices as add-ons to TurboGrafx-16 and MegaDrive, which enabled more storage space than cartridges, so games contained more videos and animations. Nintendo was also interested in adding CD-ROM to SNES and partnered with Sony to accomplish that, but Nintendo later canceled the project. After cancellation that Sony developed the project to create their home console, the PlayStation. Also, Sega's super-fast Sonic the Hedgehog [36] is a competitor for Nintendo's iconic plumber Super Mario. Handheld consoles also rushed to market during this era. Nintendo introduced a technically limited GameBoy portable gaming device with a monochromatic screen. Although more technologically advanced mobile devices, Atari Lynx, Sega's Game Gear, and NEC's Turbo Express, were on the market, they lost to GameBoy due to limited battery life and poor catalog of games.

2.5. 5th Generation of Home Consoles

The fifth generation of home consoles was the era of Sony's PlayStation which ended on discontinuation of the PlayStation. The Resident Evil [37] game series began when Capcom created its first installment to Playstation. Sega released Saturn as the successor of MegaDrive/Genesis and NEC PC-FX as the successor of TurboGrafx-16; they were no match to PlayStation. Nintendo entered the race with Nintendo 64, which had several iconic games like GoldenEye 007 [38], The Legend of Zelda: Ocarina of Time [39], and Super Mario 64 [40]. But where other devices used CD-ROM as media, Nintendo was using cartridges. The cartridges were more durable than CD-ROM and provided better cover for unauthorized copying; they had disadvantages to being more expensive and provided only 1/10 of CD-ROM storage space, which led to more companies publishing their games on CD-ROM. Atari also created a 64-bit console, the Jaguar, which was expensive and didn't have many games on its rosters, a commercial failure. Atari left the console market but is still developing games for other platforms.

2.6. 6th Generation of Home Consoles

In sixth-generation consoles, Microsoft enters the home console market with Xbox and Halo: Combat Evolved [41] as launch titles. Sony created PlayStation 2, which can run older PlayStation games. Nintendo made GameCube, which did not manage to perform as well as previous Nintendo consoles. Sega introduced Dreamcast, which had several hit games like Crazy Taxi [42], Jet Set Radio [43], and Shenmue [44]. Even though Dreamcast sales were promising, eventually they dropped, and Sega had to leave the console market and start making games for other platforms, even their old adversary Nintendo. During this era, the Internet connection was available on home consoles making network playing possible. In 2013 PlayStation was discontinued marking the end of the sixth generation.

2.7. 7th Generation of Home Consoles

Nintendo fights back to Sony's PlayStation 3 and Microsoft's Xbox 360 with Wii in seventh-generation consoles. The Wii was not able to produce HD graphics, and its technical specifications were lower than its competitors. Still, Nintendo managed to reinvent gaming by adding motion control with Wiimotes, and the Wii was the winner of the seventh generation. Sony introduced their motion-controlled playing system, PlayStation Move, which helped PlayStation 3 to outsold Xbox 360. During this era, online gaming became more critical, and Sony introduced PlayStation Network, and Microsoft introduced Xbox Live.

2.8. 8th Generation of Home Consoles

In eight home console generations, Nintendo released two different consoles, Wii U as a successor for Wii, which was not a success in the market and Switch refined some of the Wii U key concepts. Sony released PlayStation 4 and Microsoft Xbox One.

2.9. History of Mobile Games

When technology evolved and components were getting smaller and more power-efficient different kinds of devices could be created like pocket calculators and portable cassette players. The first handheld games were simple, fixed games with red Light-Emitting Diode (LED) screens and some buttons from Mattel [13] with titles like Mattel Auto Race and Mattel Football. In 1978 the first handheld game console that used interchangeable cartridges, the Microvision, was created by Milton Bradley [13]. Also, Nintendo started to develop handheld games when Nintendo's game designer, Gunpei Yokoi, saw a bored businessman punching the buttons of the Liquid Crystal Display (LCD) calculator in a train ride [45]. From that observation, Nintendo's Game & Watch series, which were simple handheld electronic games with LCD, and a couple of buttons were created. Gunpei Yokoi also designed the Nintendo Game Boy with a Nintendo R&D1 team with a directional pad (D-pad) and buttons.

Mobile phone gaming history began in 1994 when IBM released Simon Personal Communicator Simon, which contained Scramble. Siemens released S1, which had hidden Tetris [5] clone Klotz (user could start it by entering special code to the phone), and Cetelco released Hagenuk MT-2000 with Tetris clone. Where those devices were

Figure 3. Figure © Ilkka Karjalainen, screenshots from Nintendo Game & Watch Snoopy Tennis and Game Boy running the game Tetris

costly, mobile gaming was brought to the mainstream by Nokia 1997 by releasing 6110 with built-in games: Snake, Memory, and Logic. The Snake was a success and included two-player mode using two phones and an IR connection. The games were controlled with the same keypad that users entered numbers to call and send text messages. Like in the Nokia Communicator series, some high-end phones had full Qwerty-keyboard, and external keyboards were sold as accessories. In that phase, games were all pre-installed, and users can not install any more games on devices.

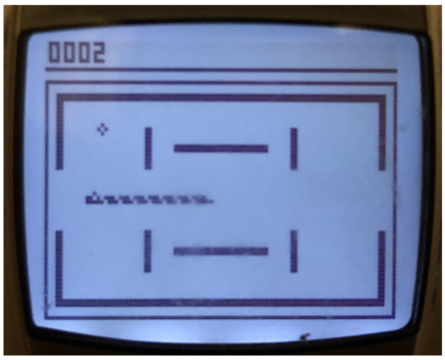

Figure 4. Figure © Ilkka Karjalainen, screenshot from Snake II played on Nokia 6310i

The option to install games by users to phones from external sources was available when mobile platforms evolved. On the Internet, sites were available where games could be bought and downloaded to mobile devices. As an example, 2003 Nokia released the smartphone N-Gage with handheld game system options. Games came as memory cards, and changing a game required taking off the battery because the memory card slot was behind the battery. The N-Gage failed, and Nokia discontinued its manufacturing. There were early massively multiplayer online role-playing game (MMORPG) adaptions, like *BotFighters*, done on mobile phones with the help of mobile operator location services, where players played with sending short message service messages (SMS) [46]. The popularity of mobile games increased when Apple opened the App Store in 2008, where users could download games and applications to devices with iOS. Google also opened their mobile games store, Android Market (later named Google Play), in the same year. With those markets, developers could develop and distribute games, and the store handles payments. After App Store and Google Play, many other companies like BlackBerry, Nokia, and Microsoft opened their marketplaces for games. When smartphone performance increased, old games like Doom could be ported to them [47].

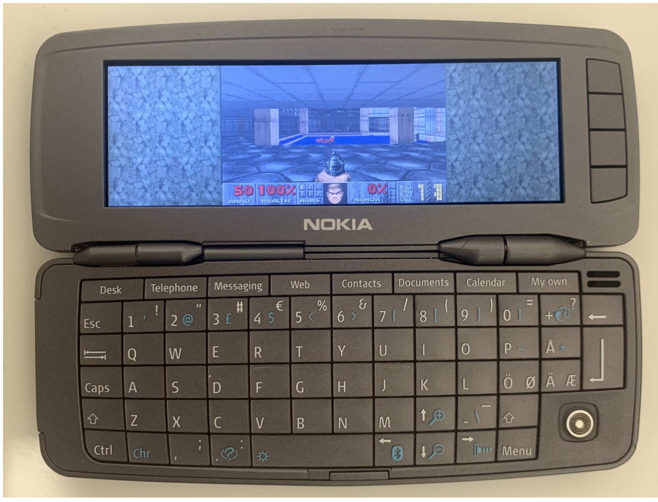

Figure 5. Figure © Ilkka Karjalainen, screenshot from Doom played on Nokia 9500 Communicator

2.10. Current Status on Mobile Games

According to the latest research [48] players use mobile phones to play games more often than other devices. Users carry mobile phones everywhere, and they enable casual, short-time gaming during spare time, when commuting, taking a bus, and even when visiting the toilet. In 2000's, and especially after 2016 release of Pokémon GO, a specific type of games became also available on mobile phones, location based mobile games and mobile augmented reality games [49]. In these games one does not just control the game by touch, buttons, or tilt, but also with how one situates on the Globe. The other gaming devices are dedicated to game playing; the mobile phones are used for other activities, like making a phone calls. The different gaming devices usually provide some controller as an interaction device between the player and the game. These controllers can have buttons, analog sticks, touch areas, and even motion sensors. The games can efficiently utilize controller functions for user actions, but this is not an option on mobile devices, since they provide only a touch screen. There has been efforts to create gaming controllers for mobile phones, but the lack of support for games has rendered those obsolete, and much of the effort has been relatively short-lived gimmicks like Mattel Apptivity Toy for iPads. Also, mimicking game controllers in touch screen providing game controllers buttons as UI elements have been tried, but that strategy occupies valuable screen space. One of the most used game engines, Unity, has many different virtual joysticks and virtual touchpad [50] on the screen operated with the player's thumbs. But of course, the virtual joystick is not mandatory for successful mobile games; some of the most played mobile games are designed to play on touch screens, like Angry Birds, Fruit Ninja and Subway Surfers.

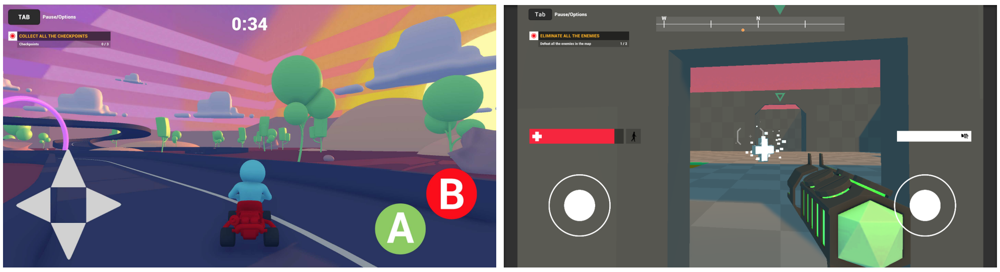

Figure 6. Figure © Ilkka Karjalainen, screenshot from virtual joysticks from Unity asset store added to Unity example games

2.11. Touch Controlling

The idea of touch controlling was first presented in a short article [51 p.219] as "This device, the 'touch display,' provides a very efficient coupling between man and machine." The main idea behind touch controlling is that the user can interact with what is displayed directly instead of the interaction device. A wide range of devices, from medical to game consoles, use touch controls. Touch controlling is in use in places where other controlling methods (keyboard, mouse, etc.) are unsuitable and places where user interactions need to be restricted only for specific actions like Automated Teller Machines (ATM). There are several ways to implement touch controlling, but two leading technologies in mobile devices are resistive and capacitive. The resistive touch controlling works so that there are two metallic surfaces, and when pushing the screen, these two surfaces contact each other, and touch position is detected. Since the touch detection is by pressure, the resistive touch controlling is not precise, so it is used in places where high tolerance is accepted. Also, pressure detection enables touching with other methods than pressing with a finger, and it is a low-cost technology. In capacitive technology, the touch position is detected by measuring the change in the surface electrostatic field when a human, as an electrical conductor, distorts that field by touching the surface. The capacitive technology is more precise than resistive, and it is thinner, so it suits mobile phones better.

Before mobile phones, touch controlling on mobile devices was made popular by Personal digital assistants (PDA), which were devices to manage personal information (calendar, phone numbers, etc.), provide access to the Internet, and sync data between PC and PDA. In PDAs there were also accessories like GPS modules which enabled location tracking and navigation services like in study [52]. There is also a tool for using touch controlling, called a stylus, which allows more precise controlling with resistive technology. Touch-controlled games can be so well designed to be intuitive that even little children can learn to play them. And when game controlling is intuitive, games can be used to educate children [53].

2.12. Motion Controlling

Motion controlling games have been mainstream since Nintendo published the gaming console Wii (and its descendant Wii U) with Wiimotes, game controllers with the direction pad, buttons, infrared sensors, gyroscope, and accelerometer. Games are played by, e.g., moving characters with Wiimotes. Also, Sony and Microsoft have enabled body controlling with PlayStation EyeToy (webcam) and Microsoft Kinetic (depth-sensing camera). Both of those required games are specially designed for them, so the main part of those platforms is not supported. The EyeToy was also used in the study [54] to understand movement as input for interaction. From a traditional game controller point of view, there are also Sixaxis and DualShock controllers available for Sony PlayStation. PlayStation also has a WiiMote clone named PlayStation move, which also requires games to be designed for it. From a historical point of view, some more obscure game controllers were created when a few are worth mentioning, like Bandai's Family Trainer and Family Fun Fitness floor mat game controllers for the original 8-bit Nintendo Entertainment System. Some Nintendo GameBoy cartridges include accelerometer sensors to control games.

2.13. Controlling with Facial-Gesture Recognition

Controlling games and applications with facial-gesture recognition and gaze tracking have been developed for people with handicaps, preventing using devices usually. One example of this kind of situation is the late Stephen Hawking, in which amyotrophic lateral sclerosis (ALS) prevented moving his hands [55]. For Stephen Hawking, the solution was to use Intel's Assistive Context-Aware Toolkit (ACAT), controlled using a muscle in his cheek. The eye controlling is also studied for behavioral therapy of Attention Deficit Hyperactivity Disorder (ADHD) in study [56]. Nowadays, this kind of technology has been enabled for Windows 10 with, e.g., Tobii eye-tracking cameras like in a study by Smith & Graham [57] where the technology was used for estimating gaze with head orientation for controlling games. Also, high-end iPhones are capable of the eye and head tracking. Apple provides an Application Programming Interface (API) [58] which are used, e.g., in Apple's FaceTime providing different kinds of effects. Also, a laptop webcam used for controlling a simple game is shown in study by Erdem et al.[9]. The eye controlling with gaze is used to substitute mouse controlling like in study by Dorr et al.[59] where a Breakout clone with name LBreakout2 was implemented or in study by Isokoski et al. [60] where chess and other games were implemented. But where the eye's gaze is used as a mouse input, eye closing is used to use as mouse click but not as a game controlling method as in this study.

2.14. Comparison of Different Controlling Methods

Games are played differently, and some kinds of interaction methods are more suitable than others. There are plenty of different joysticks ranging from iconic Atari 2600 compatible joystick Accurate Controller MK2 (TAC-2) to different kinds of racing wheels and flight controls. Also, keyboards and mouses are used to control games.

Figure 7. Figure © Ilkka Karjalainen, Different kind of interaction methods

Studies [6, 7, 8] compare head-tracking, eye-tracking, and tilt as input methods in mobile games. In [6] they found that tilt was the most accurate interaction method, Eye-tracking was the worst interaction method, and Head-tracking was between those two. The study [8] shows similar results when tilt and facial tracking were compared and found out that tilt was the better interaction method players liked facial tracking because it was an innovative and exciting way to play. In the study, [7] CameraMouse, a program that enables you to control the mouse pointer on your computer screen just by moving your head, as compared with the touchpad and keyboard. The keyboard results were the best interaction method: the keyboard, then the touchpad, and the CameraMouse.

2.15. Gaming Fatigue

Using eyes as interaction with a camera can require the player to focus on using eyes, or when using the head player might need to bend the head in abnormal positions, which can make players feel uncomfortable. Interaction with games can strain players, so to help interaction method can have controllable attributes like mouse sensitivity. Also, players prefer different interaction methods, so some games are designed to be played with varying interactions for the players to select their own preferable interaction method. Like in Microsoft Flight Simulator, players play with flight control systems, keyboard with Xbox Controller, mouse or virtual reality (VR) setup. When a player encounters too tricky a situation in-game and is forced to try multiple times to get past it, the player can become frustrated and tired, leading to gaming fatigue [61]. The game should not be too hard or too easy but provide enough challenge to keep the game flow ongoing [62]. Game developers have known about gaming fatigue for decades. One way to prevent it is to allow players to choose from different difficulty levels, from easy to hard, to match players' skill levels. When the player's skill level increases, there is a possibility to play the same levels again with more challenges. Some games also reward players for completing a game with the most challenging skill level by providing another, even more, difficult skill level or awarding achievements.

Figure 8. Figure © Ilkka Karjalainen, screenshot from skill level selection in game Wolfenstein 3D, developed by id Software.

Gaming fatigue can be prevented by designing better interaction methods to play the game but also with adaptive difficulty. Some games contain a tutorial section that teaches players how to play the game to ensure they know how to play the game. Another way to prevent gaming fatigue is to use dynamic difficulty [63, 64]. Modifying game difficulty during gameplay can be made with several different methods [64]. In many racing games, method names such as rubber-banding are used where the opponent's vehicle speed depends on the player's success. If the player is succeeding, the opponents' vehicle speed is increased and vice versa. More subtle methods are also used, like in the upcoming Left 2 Dead 3 [65], where an AI Director approach is used. The AI Director, which is also called procedural narrative, analyzes a player's playing attributes like how fast the player is moving, shooting accuracy, etc., and generates new situations and challenges. Hence, every gameplay time is different. In-game Max Payne, the dynamic difficulty appeared so that the level of aim assistance increased when the player is doing poorly and increased enemy health a bit when the player is doing very well. This adjusting the game to match the player's skill aims to get the player to a flow state [66] and and provide a more engaging immersive player experience, so the player continues playing. One classic method for keeping players interested in playing the game is to use procedurally generated levels so that no game is the same as a previous one. The 1980s hit game Rogue and its descendants use that approach, and they have kept players playing for decades even though the graphics are just ASCII characters.

Also, computers and gaming consoles have become more high-powered games with an open world. With open-world games like GTA V [67] where players can complete different scripted tasks but, in addition to that, they can explore an open world. Sandbox games are games where the player can complete a task with several different ways to reach the goal, and some of the sandbox games do not have a specific goal. Or when a goal is reached, the game restarts from the start like in game No Man's Sky [68]. Some games, like Minecraft [69], enable modification of the game itself with scripts and custom challenges that can be created by players.

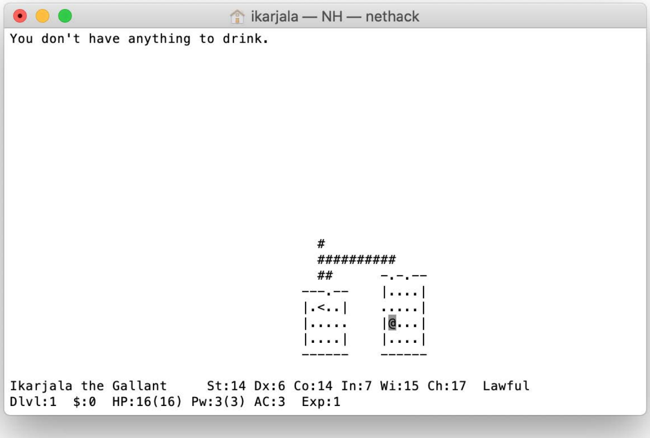

Figure 9. Figure © Ilkka Karjalainen, screenshot from roguelike video game, NetHack played on modern MacBook.

2.15.1. Gaming Fatigue in Mobile Games

There are millions of games to choose from in the current mobile market, so gaming fatigue becomes an essential part of game design. It is so easy to remove a game and install a new one that gameplay should be so intuitive that playing games should be possible straight away. If the game is not intuitive enough, players can easily decide that it is too complicated and proceed to the next game. Also, non-game-related issues may lead to gaming fatigue in mobile games like ads and microtransactions. When the game is stopped due to an ad, and the player needs to wait for the ad to finish player might get frustrated and uninstall the game. The same can happen when the player sees that money needs to be spent on microtransactions to advance. According to AppsFlyer report, [70] about half of those games are uninstalled within 30 days of initial installation.

2.15.2. Gaming Fatigue with Different Controls

Joysticks have been used in gaming since the 1970s first mainly in arcade machines, home computers, and home consoles. But joysticks are also used in airplanes where fatigue can cause disasters, so the joystick caused muscle fatigue is studied in connection with planes [71] more so than in gaming. There are many different types to control games, and some game controllers are created for use only in one game. Some game controllers have an analogy to real world objects like electronic light guns. One of the most memorable light gun is Nintendo Entertainment System Zapper [72]. Zapper and other light guns share the same fatigue effect: tiredness to hold a weapon when it is aimed at the screen. Just using the mobile phone can cause fatigue to the thumb and wrist [73]. In mobile games, physical game controllers can also be used, but most mobile games are touch as interaction methods. In the study Abbaszadegan et al. [6], custom game TrackMaze is created with three different interaction methods: eye-tracking, head-tracking, and tilt. Participants were presented after playing with all interaction methods; a questionnaire revealed that eye-tracking caused more fatigue than head-tracking, and the least fatigue was caused by tilt.

2.15.3. Research Methods on Gaming Fatigue

Gaming fatigue is difficult to research because it is about players and games. Each player has different preference for games and control setup. For example, there is no end in the classic game Tetris [5], and the game will continue as long the player can keep on playing, so the score is the way to rank players. Every Tetris game repeats the same pattern, and some players do not like that where others can not stop playing Tetris, and there are even world championship events arranged yearly. Some players play online first person shooter (FPS) and sports like hockey and football games for the social aspect of playing. So, to get valid results gaming fatigue should be tested on game types that research subjects like to play most. For example, if the player likes to play Tetris, he or she should be used as a research subject when the game to be tested with has similar elements like Tetris to get reliable results that may still not be generalizable to other game genres. Players might get bored for other reasons than fatigue if the game does not offer a suitable way of playing like described in study [74]. Player feedback is needed to research gaming fatigue. Game developers can get feedback from players via Internet communication channels and patch their games to make them more playable but more scientific methods [75]. In mobile games, different kinds of software development kits (SDK), like Google's Firebase, can be used to collect information on how players are using the game. This kind of automatic information collecting during game playing is called game analytics, where player behavior is analyzed with tools and metrics. Game developers can add traceable signals to code when, e.g., the player starts to level, the player dies, and other essential aspects in the game. With those signals information about how much time players took to finish the level, which level takes many retries, which level was the last before the app is uninstalled. Of course, it is hard to pinpoint even with that game analytics and player tracking what causes players' frustration. In the study by Yun et al. [76], particular setup, StressCam, was created to monitor players during gameplay. The design contains a camera that monitors the player's facial physiology. By inspecting blood flow in the supraorbital region in the player's face, game difficulty is dynamically changed to keep the player's flow ongoing, so frustration is reduced, and the player keeps on playing. In a the study by Nosu et al. [77], the same kind of particular setup was used. Setup detects facial expressions instead of dynamically changing the game difficulty. The system provides verbal feedback messages to help players navigate. Also, EEG can determine players' frustration to adjust game flow [61] dynamically.

3. IMPLEMENTATION

When developing a game, the development seldom starts from a scratch. There is a great number of game engines to choose from, and certain criteria had to be met when I chose an engine for this study:

- Sample code and tutorials available
- Open source
- Extendable with native programming language use mobile phone hardware (camera, accelerometer) to enable all interaction methods. Some features needed might not be available in the engine and need to be implemented during this study
- Support for a physics engine

Sample code and tutorials were needed to reduce implementation time. Open source engine was favoured so there would be a possibility to modify the source code to fit the user study needs better. If there would have been no support in the game engine for using mobile phone APIs for facial camera recognition, then a plugin would have been needed to be written to enable that. This implementation uses a physics engine to simulate ball movement, so the game engine needed to support physics.

Table 1. Evaluated game engines for study implementation

Engine	Sample code	open source	extendable	physics
Cocos2d-x	Yes	Yes	Yes	Yes
Unity	Yes	No	Yes	Yes
Unreal Engine	Yes	No	Yes	Yes
SpriteKit	No	No	Yes	Yes
Godot	Yes	Yes	Yes	Yes
libGDX	Yes	Yes	Yes	Yes

Chosen game engine for the study implementation was Cocos2d-x [78]. Several game engines fulfilled the criteria and were good candidates. Still, the reasons for choosing Cocos2d-x were good extendability for using mobile phone's native APIs, support for Box2d physics engine, and support using C++ as a programming language. Even though the support for using C++ as a programming language was found during the evaluation, the implementation time was reduced due to my experience and expertise in C++. Selection of a mobile phone platform for the study was between Apple's iOS and Google's Android. They both are provided for mobile phones which have the capability to recognize facial gestures. Both platforms offer what was needed to implement the research game, and I have developed applications for both platforms. The choice was made to implement for iOS first and, if there was still time, also implement Android-specific parts to make the case study available for both platforms. The iOS was chosen to be first since I had iPhones available.

3.1. The Research Game: DippaOut

For this study, a game called DippaOut was implemented. It joins a line of Atari's Breakout [21], presented in Figure 10, successors. The game contains the same idea of physics, where the ball moves around the game area and breaks blocks. When all the blocks are removed, the level is complete, and the player can advance to the next level. When the block is hit buy the ball, it ricochets in another direction with the same angle where the hit occurs. Still, when the player hits the ball with the paddle, the speed and angle are altered by the paddle's part where the collision occurs and the speed and direction of the paddle. In the original Breakout, the player controls a paddle, which is used to prevent the a from dropping from the game area, and also, the paddle is used to aim to block. Aiming in Breakout's is quite vague since ball movement follows Newton's laws of motion, and the direction of the ball is altered by collision to level walls, blocks, and the player's paddle. My implementation extends the original Breakout idea with the same elements as its successors, like the Arkanoid [79], by adding multi balls, different kinds of blocks, etc. The DippaOut adds its twist to the game by removing the player paddle from the game and adding a new type of controlling method, the level itself. The level rotates, and the player needs to use blocks to prevent the ball from going off the game area. Also, the game area is changed from a rectangular shape to a circle shape, so it is fully visible all the time, and controlling is more intuitive. And where in original Breakout, there are lots of rectangle shapes where the ball hits, which makes the ball movement predictable, but in DippaOut, the shapes are curved, so ball direction is hard to predict. The name of the game, DippaOut came from a strange coincidence with Breakout and my own life; the background story of the Breakout is about a prisoner who tries to escape from prison; this study helps me to finish my university studies (Diploma in Engineering, which has nickname "*Dippa*" in Finnish)

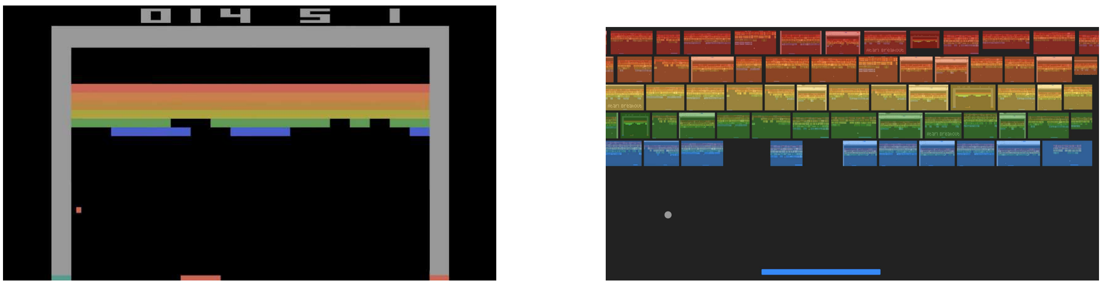

Figure 10. Figure © Ilkka Karjalainen, screenshots from Breakout played on Atari 2600 and Google's Easter egg Breakout from 2013

3.2. Box2D

Box2D [80] is an open-source 2D physics engine for games by Erin Catto. For the DippaOut, the Box2d was chosen to be the used physics engine because it provides continuous collision detection and contact callbacks. Box2d is implemented with C++, which reduced implementation time because I could use my C++ skills for implementation. The physics engine is utilized for ball movement because the ball is a dynamic body that collisions with the arcs, which are static bodies [81]. The collision detection informs when the ball has begun to touch the arc. The arc-specific action is run depending on the arc type. Also, since the ball is a dynamic body its speeds decreases during the time because of friction and collisions; when the ball collides to arc, its speed is stored, and after a collision, the stored speed is applied to the ball in an inverse direction. Several control methods were implemented for DippaOut to be able to use the game in a user study comparing these methods.

3.3. Control Method 1: Touch

Touch controlling is the traditional way to control mobile phone games. In this study, the ball movement can not be controlled directly by touch, only rotating the game area clockwise or counterclockwise. When the player touches the mobile phone's screen, the angle of the point of touch and the center of the game area. When a player moves touch to another point, the angle is calculated again and compared to the previous angle. If the new angle is more bigger than the previous angle, the game area circle is rotated clockwise and vice versa. The touch detection is handled with Cocos2d-x touch detection APIs, which are basically wrappers to platform-specific touch API.

Figure 11. Figure © Ilkka Karjalainen, screenshots from simple level play through by moving level left and right

3.4. Control Method 2: Rotation

Controlling the game by rotating the smartphone is done with the help of the gyroscope, which is an instrument that senses rotation by its own without the need for any other instruments [82]. In smartphones, gyroscopes provide information about the device's orientation in three-dimensional space. The smartphone attitude (rotation) is reported from three different axes: pitch, roll, yaw, and values from those axes reported as radians ranging from π to -π. In the DippaOut, we are only interested only in the yaw axis. Cocos2d-x does not have built-in support for gyroscope, so a native wrapper was needed to get native gyroscope functionality from smartphone OS. The gyroscope wrapper creates an instance of native gyroscope manager in iPhone, which reports changes with predefined interval (the interval is configurable and has been set to 500ms). When the gyroscope has been received from the native side, it is delivered to the Cocos2d-x side with the help of the Cocos2d-x event dispatcher system to ensure thread safety and synchronicity between the native and Cocos2d-x.

To get information on how much the smartphone is rotated, a so called reference attitude frame needs to be configured. After the reference attitude is configured, one can calculate the difference between it and later reported smartphones attitude frames. That calculation is the difference between these two attitude frames, which tells the pitch, roll, and yaw values as radians. With that result, the level of the DippaOut rotated. The rotation is also applied to ball movement, moving in the same direction when the phone is rotated. Figure 12 shows how the level and ball are moving when the phone is rotated clockwise 180°. First, the ball is moving northeast to the red arc direction, but when the phone is rotated 90° the ball continues moving to the northeast, but now the level is rotated so that the green arc is now where the red arc was before. When the phone is rotated another 90° the green arc has been rotated away, and the ball continues to head northeast but now towards the blue arc.

Figure 12. Figure © Ilkka Karjalainen, screenshot from when the smart phone is rotated the ball moves all the time in the same direction and the level rotates around the ball.

3.5. Control Method 3: Eye Closing

One of the ways of controlling DippaOut is done with smartphones that can recognize facial gestures. The standard smartphone camera is not enough, and other equipment is needed. In Apple's high-end iPhones, starting from 2018's iPhone X, facial gesture recognition is done by a dot projector that projects infrared dots and infrared cameras which then take an infrared image. Also, there is a flood illuminator that adds more infrared light when needed. According to Apple, the infrared output is so low that it does not harm the user. With that, infrared information depth-awareness is added to the front camera feed. The system in iOS devices TrueDepth camera and is used, e.g., in FaceID, biometric authentication for unlocking a device. The TrueDepth camera is not available on all iOS devices, so in DippaOut, there is a need to check first if all needed features are available. If that check is successful, the TrueDepth is started. Before DippaOut can use the camera, user authorization is required because, in iOS, the user must explicitly grant permission to access the camera.

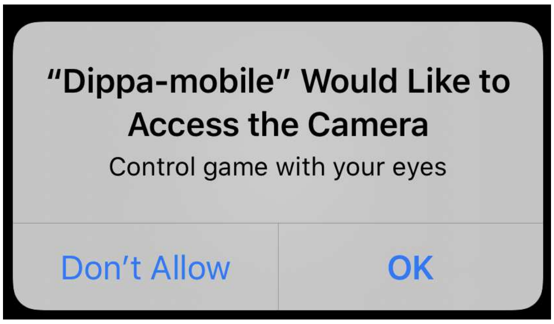

Figure 13. Figure © Ilkka Karjalainen, screenshot from DippaOut asking permission to use front camera

The TrueDepth camera is the hardware part; the ARKit [83] is the software part of facial gesture recognition. The ARKit provides an API to detect multiple faces and track their position and orientation. ARKit creates an object that represents face topology where AR features like emojis and makeup can be added. The face dimensions and facial expressions are also detected.

Controlling DippaOut by closing eyes is done with the help of the ARKit, which provides information about eye status with BlendShapes. BlendShapes represent a movement of specific facial features like smiling, closing eyes, etc., recognized by ARKit. The market reports the eye status with values from 0.0 to 1.0, where 0.0 means that the eye is entirely open and 1.0 completely closed. The threshold between open and closed eyes is configurable in DippaOut to enable user-specific settings. When controlling DippaOut with opening and closing eyes, the same mechanism is used when controlling by touch. When both eyes are open, the level stays in its place. If a player closes the left eye, the level is rotated counterclockwise, and when the player closes the right eye, the level is rotated clockwise. If both eyes are closed, the level is not rotated at all.

3.6. Control Method 4: Head Tracking

The player's gaze has been utilized on virtual reality (VR) games to provide deeper immersion in-game and the ability to browse game worlds from different perspectives. In VR, glasses provide information for games where the player is looking at. In DippaOut, Head Tracking is done with the help of Apple's ArKit [83], which has face-tracking built-in. The phone tracks the face and rotates the level to where the player's head is pointing at.

3.7. Control Method 5: Eye Tracking

There is no specific API available in ARKit to tell where the phone's screen user is looking. Looking towards the screen is detected with projection. The ARKit only provides information about the center of the eyeball and z-axis from the center of the eyeball in the direction of the pupil. Those rays can be drawn from the center of the eye in the direction of the user's gaze. At the end of the ray, a dot is attached. The ray is attached to the eye's position and direction, so when the user moves the eye, the ray moves correspondingly. When the ray's position is updated, the dot's position in the world is projected to screen with renderer (which draws the current image). The projection result tells the position of the dots on the screen. The projection is needed for both of the eyes, and the average is calculated, which provides an estimation of where the user is looking at. Of course, if one eye is looking left and the other is looking right, this method will not work, but that is acceptable because doing that kind of trickery makes the DippaOut unplayable is not looking at the screen of the phone. The difference between controlling Head Tracking and Eye Tracking is that in Head Tracking, the head movement is allowed, and with moving eyes, the head should be stationary.

3.8. Starting the Game

In DippaOut, there are three different game start sequences. The first sequence is a simple one-second "GO" text animation with fading from visible to invisible, and after that, the game starts. This sequence is used in Touch, Rotate, and Eye Closing. The Head Tracking interaction has a custom start sequence to ensure that gameplay is as smooth as possible. To make sure that the player is facing the camera and head movement is detected correctly player has to rotate head 360° clockwise, and after that game starts. Also, Eye Tracking has a custom start sequence for the same reason as in Head Tracking. In this sequence, a player should concentrate on eye focus, which is represented as a white circle to the center of the screen, where is a red circle. Those white and red circles should overlap each other for one second, and after that, the game starts.

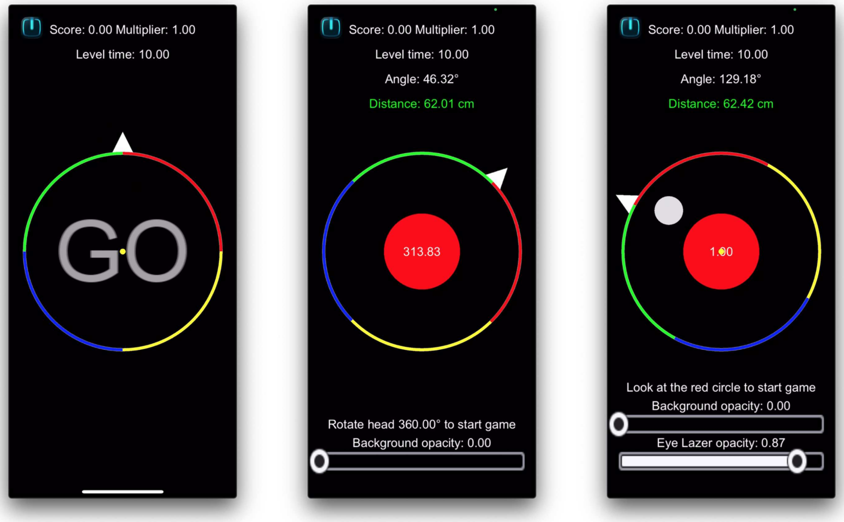

Figure 14. Figure © Ilkka Karjalainen, screenshots from different game start sequences

3.9. Settings

To reduce gaming fatigue caused by too challenging gameplay, the DippaOut provides configurable settings to find the most playable attributes. When the best configuration was found after testing, this configuration was used for the user study.

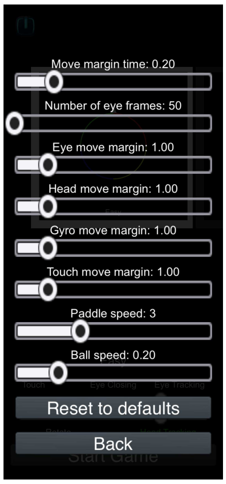

Figure 15. Figure © Ilkka Karjalainen, screenshot from DippaOut settings.

4. STUDY PLAN

Mobile phones lack a physical joystick, and they need to provide other possibilities to control games, like cameras, an accelerometer, and a gyroscope. The games need to be redesigned to use different interaction methods provided by these devices instead of a joystick or touch. This thesis implements a game that offers three other interaction methods, which are: touch control, motion control, and camera control. The camera control part provides three different interaction methods, gaze recognition, where the orientation of the face detects the user's gaze, and eye control, where the user's eyes look at the point on the mobile phone screen. Eye control, where controlling is done by closing the left or right eye. This research game implemented during this thesis is a simple, physics-based game with simple control. It is described more in detail in the Chapter 3. Implementation. The game DippaOut pays homage to one of the first arcade games, Atari's Breakout, and its successor, Super Breakout [84]. The idea in Breakout is to clear the level from blocks by hitting them with a ball and preventing the ball from dropping from the bottom of the game area with a paddle, which is controlled to left and right by the player. Super Breakout is the same game as Breakout but adds different game modes.

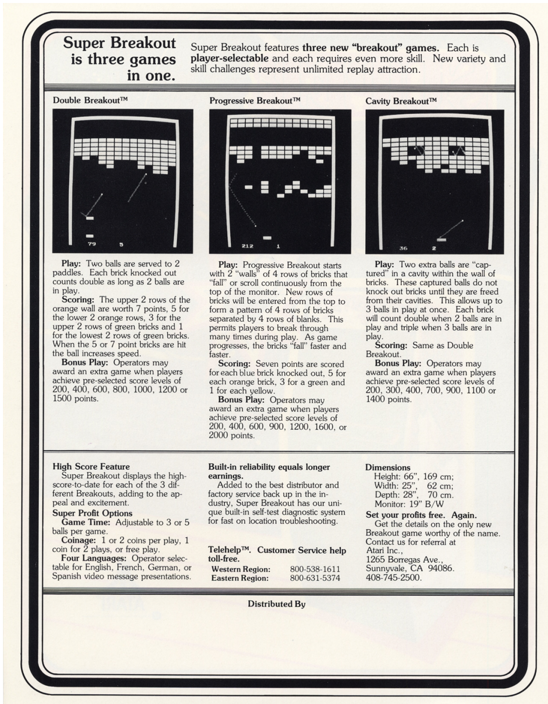

Figure 16. Figure: Public domain, the scanned version of the original flyer of Super Breakout

DippaOut, is used to carry out several tests of different kinds of game controlling methods to compare their player experience. There have been similar studies like in [85] where accelerometer and touch-based inputs were compared in mobile games, and touch-based input produced better results than an accelerometer. Newer studies also compare head-tracking and eye-tracking to tilt as in study by Abbaszadegan et al. [6] where tilt was the best controlling method. This study follows the same structure as both [85] and [6]. Those studies define their metrics to compare interaction methods and use game testing platforms.

4.1. Game Analytics Metrics

There is an inbuilt data collector for game analytics implemented in DippaOut, which collects information about player actions when playing. Information about the player's interactions with different interaction methods, game playing time, success, etc. After each playing session, collect data is stored in a testing device as XML for further processing. When the player finishes the level, a questionnaire is presented about the played session, and these answers are also stored in the device. After the testing session, another questionnaire is given to the player, where information about the player and the player's personal opinions of interaction methods. The following game analytics metrics are collected during the play session.

4.1.1. Level Completion Status

In level completion status, the end status of the game is collected. The status is either success or failure. Within that information, a count of played games and also success/failure ratio is calculated to provide results which interaction methods need more tries to complete successfully.

4.1.2. Circle Movement

As the study by Natapov et al. [86] states, "The main measure of comparison of different input devices is throughput" when computerization was done with a standard Xbox gamepad and trackball controller. Throughput measures the rate at which something is processed, and it is usually measured in bits per second (bps). The Circle Movement is a metric that calculates how many degrees the circle has been rotated in clockwise and counterclockwise and how much the circle is rotated at the time. This will result in angles per second where a lower rate is better. The count of how many times the player has moved the circle in either direction is also stored.

4.1.3. Level Completion Time

The level completion time is the same metrics used in the study by Abbaszadegan et al. [6] where head-Tracking, eye-Tracking, and tilt were compared. Also, in study by Cuaresma et al. [8] tilt-input and facial tracking as input were compared, and survival time in-game StarJelly was one of the metrics. The level completion time measures the time how long it takes for the player to finish the level. The time is measured as milliseconds. The goal is to compare the level completion time for each interaction type. The circle movement is also stored when the level is completed. Analysis on completion with different interaction methods can be done with circle movement.

The level completion times should be constant with all the interactions because the game mechanics (count of arcs, ball speed) stay the same and only the interaction type changes. The circle rotation speed depends on interaction methods, e.g., rotating the phone is slower than opening and closing eyes, so level completion time examines whether the interaction method affects how fast the player completes the level.

4.1.4. Level Score

The level score depends on hitting circle arcs with the ball. Executing combos (e.g., hitting multiple same color arcs in order) gives extra points. To achieve a top score completing the level is not enough. A more strategic approach is needed. That more confident that player is with interaction method that more strategic approach can be taken and score more points.

4.1.5. Filtered Movement

In some interactions, the movement's beginning and the end is not precise e.g. in Rotate where gyroscope is used, the testing device reports device orientation updated all the time even if the player tries to keep the device steady all the time. This constant update of the device orientation can render the data unusable because those events are not player initiated. To reduce the risk, these unwanted movements need to be filtered out to provide correct information. The default behavior is to filter out moves that are less than 1° but can be configured. The Eye Closing is the only interaction method where there is no similar filtering because that movement can be determined precisely. The Touch interaction also contains filtering because the player can rotate the circle in both directions without taking the finger off the device screen. If touching the device screen was used to detect start and end circle rotation, we would lose the information if the player changes the rotating direction without taking a finger from the device's screen. The filtered moves are logged during playing and analyzed to tell how precise the interaction method is. With that data, interactions can be compared to each other to see which interactions have the most filtered movements. The fewer filtered movements the better players have managed to rotate the circle.

4.1.6. Failure Arc Count

When the level is finished unsuccessfully, there must be some arcs left. In this game analytics metric, the count of arcs left when the level is finished is stored. With this more detailed comparison within an interaction, methods can be made. The failure count might not provide information to distinguish interaction methods from each other. With failure, arc count can be inspected how close the player was to finish a level successfully so that the lower is the failure arc count, the easier the interaction method is.

4.1.7. Between Games Questionnaire

Psychometric questionnaires are used in research and game companies to test game usability and player experience [87]. After finishing the level successfully or unsuccessfully, a questionnaire about the current level and interaction method is presented. Each question is answered with a slider with Likert scale from value one to seven, where one strongly disagrees and seven strongly agree. Four is conspired as neutral. There are thirteen questions in the questionnaire. The slider was used to gather answers because the player has to answer questionnaires fifteen times (five interaction methods, three levels), and asking for written feedback could exhaust the player's motivation for testing.

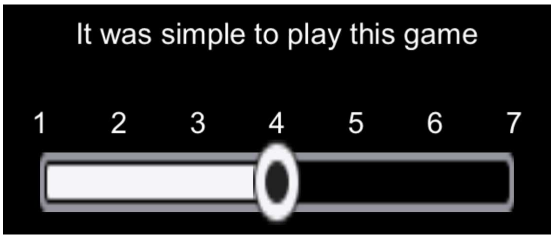

Figure 17. Figure © Ilkka Karjalainen, Screenshot from end level questionnaire slider with default value

In study [88] presented ways to evaluate usability with psychometric questionnaires. The scale by Barnett, Gatzidis & Harvey [89] on usability and first time user experiences with mobile gaming was adapted to be used in this study. Two questions from the five item dry eye questionnaire [90] was used because eye controlling is amongst some of the interaction methods.

Table 2. Between games questionnaire

Question
Overall, I am satisfied with how easy it is to play this game
It was simple to play this game
I could effectively complete the objectives and challenges
I was able to complete objectives and challenges quickly"
I was able to efficiently complete objectives and challenges
I felt comfortable using this system
It was easy to learn to play this game
Whenever I make a mistake in the game, I recover easily and quickly
The organisation of information on the game screens is clear
The interface of this game is pleasant
I like using the interface of this game
While playing the game my eyes felt excessively watery
While playing the game my eyes felt dry

4.2. Postquestionnaire

After completing all gameplay sessions, the player was asked to fill a postquestionnaire. The postquestionnaire's purpose is to get feedback from the players' subjective player experiences, which may differ, if for example playing with rotating is more difficult than playing with touch, or it might be more challenging and enjoyable. The postquestionnaire also contained a demographics section and a section where players' player experience about interactions,e.g., were asked. The list of questions was:

- Player age
- Is player eyesight normal
- How many years players have played
- What kind of gamer player is
- What platforms player uses for playing
- What kind of games player plays. There is no complete list of game categories, so using [91] was agreed to be used
- What kinds of interactions players uses when using a phone
- How the players are using the camera of the phone
- The players is asked to rate all interactions from the best to the worst
- players has allowed free word about interactions and testing session

4.3. Test Setup

Levels used in testing are designed to turn gradually more difficult from the first level to the last. This will provide a learning curve to players to reduce frustration while providing challenges when completing levels. The gameplay, design, and elements stay the same at each level, and only the size of the circle arcs (as degrees) change. When starting the level, the direction on the ball is generated randomly from 0° to 360° degrees.

Table 3. Test level attributes

Name	Arc count	Single arc angle	Time
Easy	4	90°	10 sec
Medium	8	45°	18 sec
Hard	12	30°	26 sec

To facilitate more skill-based playing, there is a Score Multiplier in the game. The Score Multiplier doubles points when the player has managed to hit two or more same color arcs in a row. When the arc hit is different color than the previous hit arc, the Score Multiplier is reset. The Level time runs downwards when the game starts, and the time left after the level is cleared added to the score as a bonus time.

The user study was done during the first quarter of 2021 when Covid-19 restrictions were in effect, so there was only observer and player present. The players were advised to speak out and give comments during testing. Think aloud protocol [92] was used,

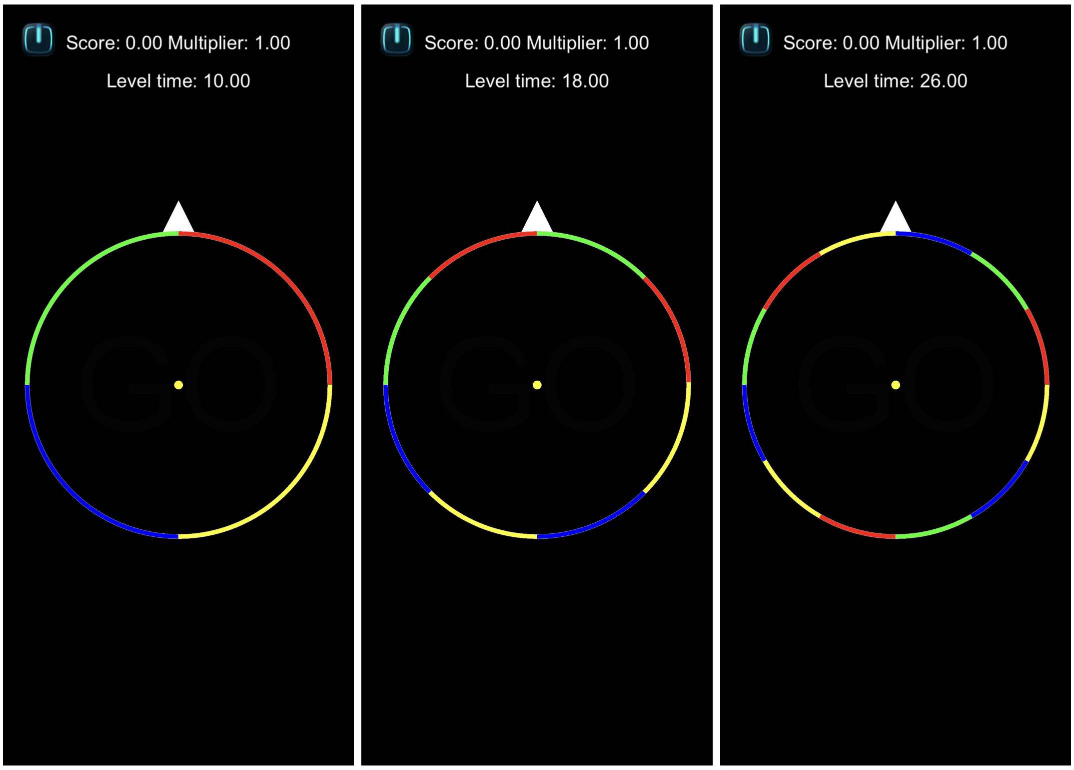

Figure 18. Figure © Ilkka Karjalainen, Screenshots from test levels: Easy, Medium and Hard

and the player was also able to ask clarifying questions from the observer. The player was also given four pages long manual on how to play the game and follow the test procedure.

4.4. Testing Environment

The testing was done in a peaceful office and meeting rooms to exclude distractions. The rooms where testing was done were well lit to provide optimal performance options for the camera. The windows of the rooms were also closed so the player's eyes wouldn't wander from the phone's screen to other interesting things because that would affect the player's performance.

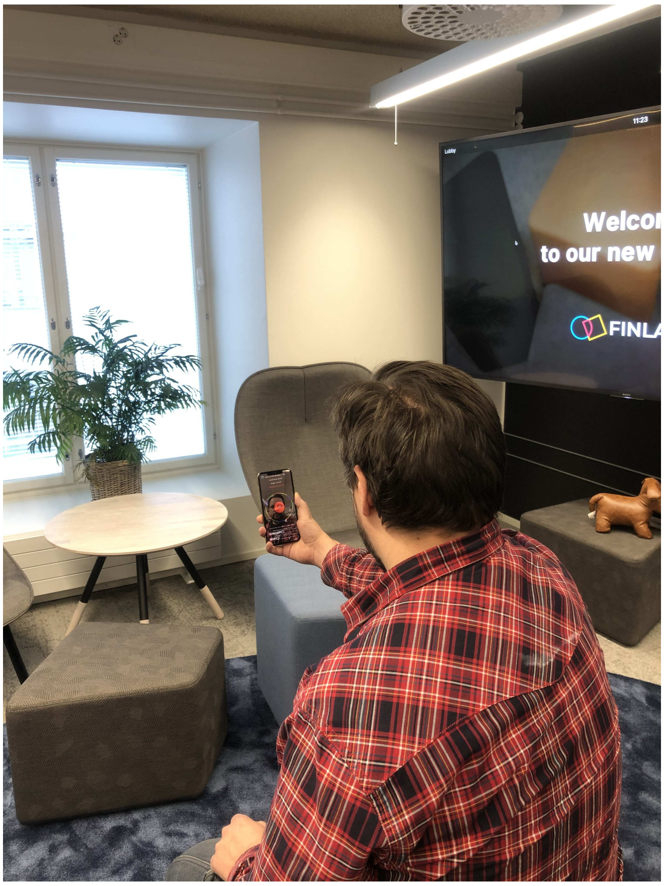

Figure 19. Figure © Ilkka Karjalainen, Screenshot from player playing DippaOut

4.5. The Gaming Device

The apparatus used during the user study was Apple iPhone X running iOS 13.5 with 7MP TrueDepth camera. Before each test session, the iPhone was reset, and the DippaOut was reinstalled.

4.6. Data Processing

The DippaOut logs anonymously players' actions and circle movements, and after each game session, that information is written to extensible markup language (XML) in the device alongside the questionnaire results. That XML is transferred to MacBook, where the XML is processed with Python scripts to stored results in SQLite database. With the database, the data is then refined to results with structured query language (SQL) commands. The statistical analysis of the results was done with Microsoft Excel Version 16.48 with Analysis ToolPak.

5. TEST RESULTS

Players played all the levels with the same setup (ball speed, circle rotating speed, etc.), and the only variables between gameplays were the interaction method and the level. The 17 voluntary players were my colleagues at Finlabs [93] and my family members, ranging from 16 years to 52 years (mean = 34.1, SD = 10.5). The players were recruited from people I would have met during the time of testing anyways to accommodate the social distancing measures in place during the COVID-19 epidemic 2021. Tests were conducted at Finlabs office and my house in well-lit rooms so camera usage would work as intended. The years played also reflects the significant age difference because years played ranged from 10 years to 36 (mean = 22.2, SD = 10.4). From seventeen of players, eleven had some problems with eyes (players mainly suffered from farsightedness, nearsightedness, and astigmatism). Fifteen of the players played 15 hours a week or less, 2 played more than 15 hours and up to 30 hours a week, and no one played more than 30 hours a week [94]. Figure 20 shows the players are the most familiar with playing games on mobile phones by touching the screen, and about a quarter have played games with rotating and facial controlling is the least known. All the players were familiar with using cameras in mobile phones and multiple different ways as shown in Figure 21.

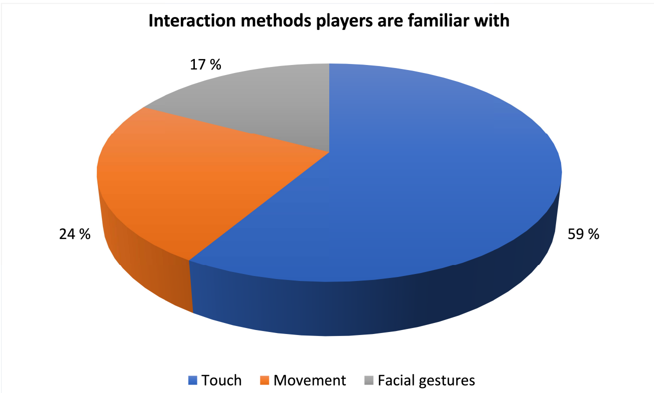

Figure 20. Figure © Ilkka Karjalainen, interaction methods players were familiar with

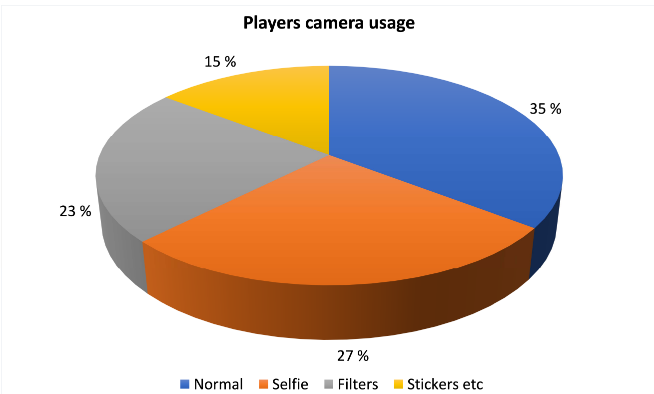

Figure 21. Figure © Ilkka Karjalainen, camera usage players were familiar with

Breakout [21] was created on an arcade platform which is relatively scarce in modern days. Players were asked which gaming platforms they are familiar with. PC, console, and mobile phones are pretty evenly used platforms that players use for playing but not so many have not been using a handheld console as seen in Figure 22. The lack of playing handheld consoles might explain that the mobile phone is used for casual playing instead of, e.g., the Nintendo DS family.

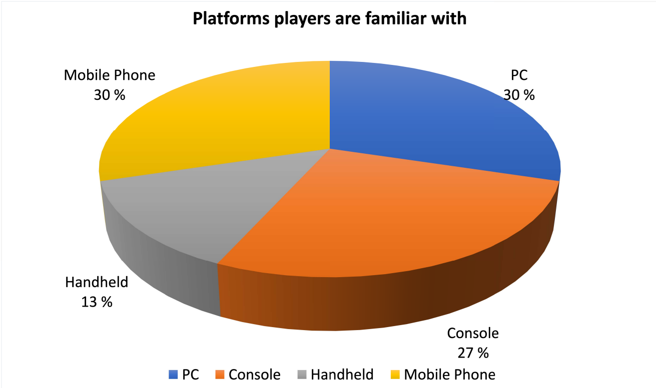

Figure 22. Figure © Ilkka Karjalainen, platforms used by players

Breakout [21] as well as DippaOut is a skill-based action game, so players were asked about their gaming preferences to see how familiar Players are with action games. If action games would be the only games that players play, the results of measurements could be biased due to their expertise. The players were given a list of different game categories where they could choose as many game categories as they wanted. When asking what kind of games players play, the players play quite a variety of different game types as seen in Figure 23, but it seems that shooting, adventure, and strategy games are the most played ones by the players. There were at least a couple of players in each category, but fighting, simulation, and uncategorized miscellaneous games were the least played by the Players.

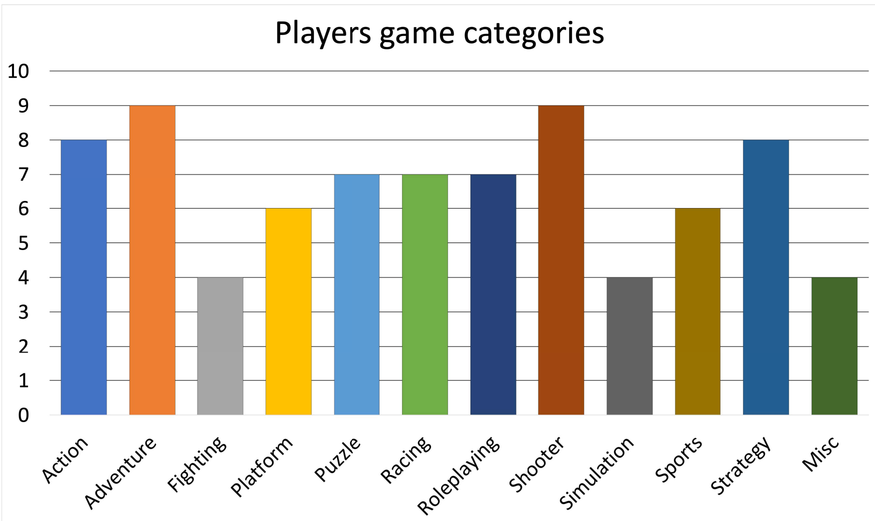

Figure 23. Figure © Ilkka Karjalainen, players game playing categories

5.1. Interaction Comparison by Players

After testing, players were given an opportunity to order different interactions on a scale of one to five, where five were given to interaction, which was the most suitable interaction for playing DippaOut on players' opinions. Figure 24 shows that the Touch is the best interaction and Rotate the second in players' opinion. Eye Closing was rated as the third-best interaction, Head turned the fourth, and Eye Tracking is ranked as the worst interaction method.

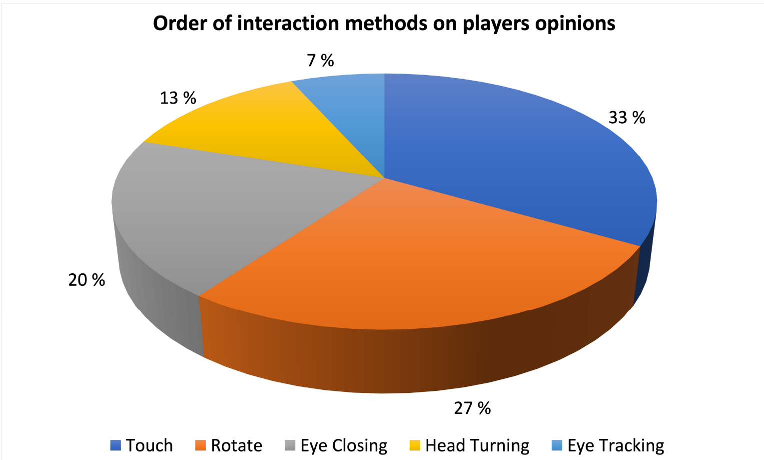

Figure 24. Figure © Ilkka Karjalainen, Order of interaction methods rated by the players

5.2. Level Completion Status Results

The Figure 25 shows the level completion status of interaction methods of different levels. This is the simplest metric to compare interactions because it adds how many times each level was played with each interaction successfully and unsuccessfully. From the total count of games, a percent of successfully and unsuccessfully finished games is calculated to make comparison easier.

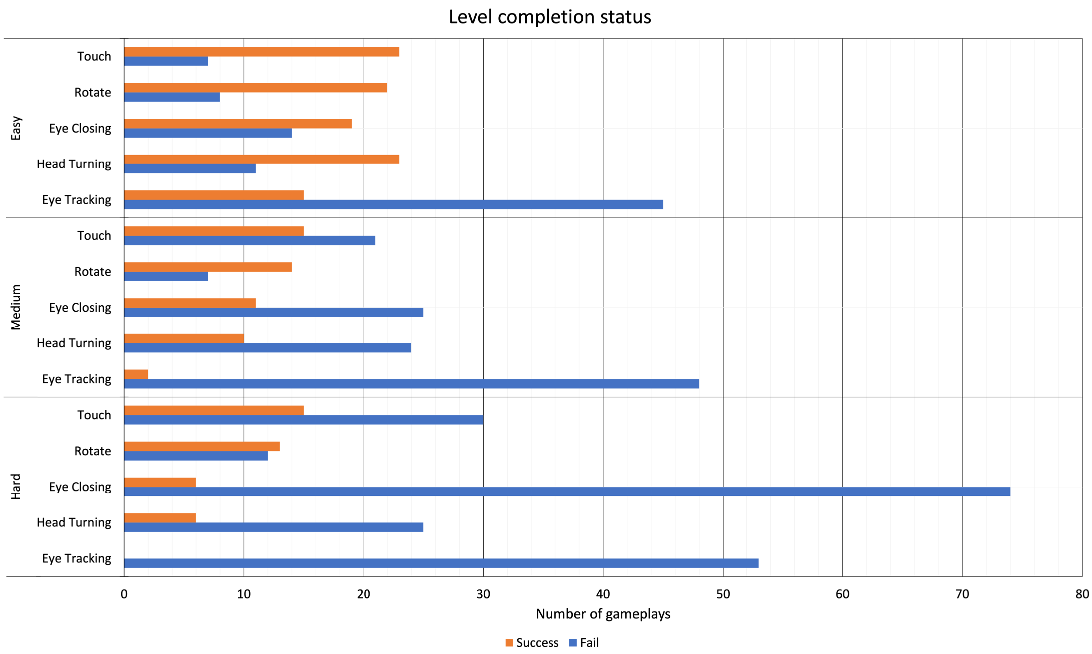

Figure 25. Figure © Ilkka Karjalainen, level completion status results

5.3. Level Score

Figure 26 shows the results about the scores that players could get when finishing the level successfully. Microsoft Excel's Single Factor ANOVA was used to the data and there were no statistically significant results because all p-values were $p < 0.05$. The ANOVA provides a probability value (p-value), which tells whether the results happened by chance. In level Easy, Medium and Hard works, these p-values were too high (Easy $p = 0.623$, Medium $p = 0.481$ and Hard $p = 0.082$, Appendix 1 : Table 12), so there is a possibility that their results occurred by chance, so they are not statistically valid.

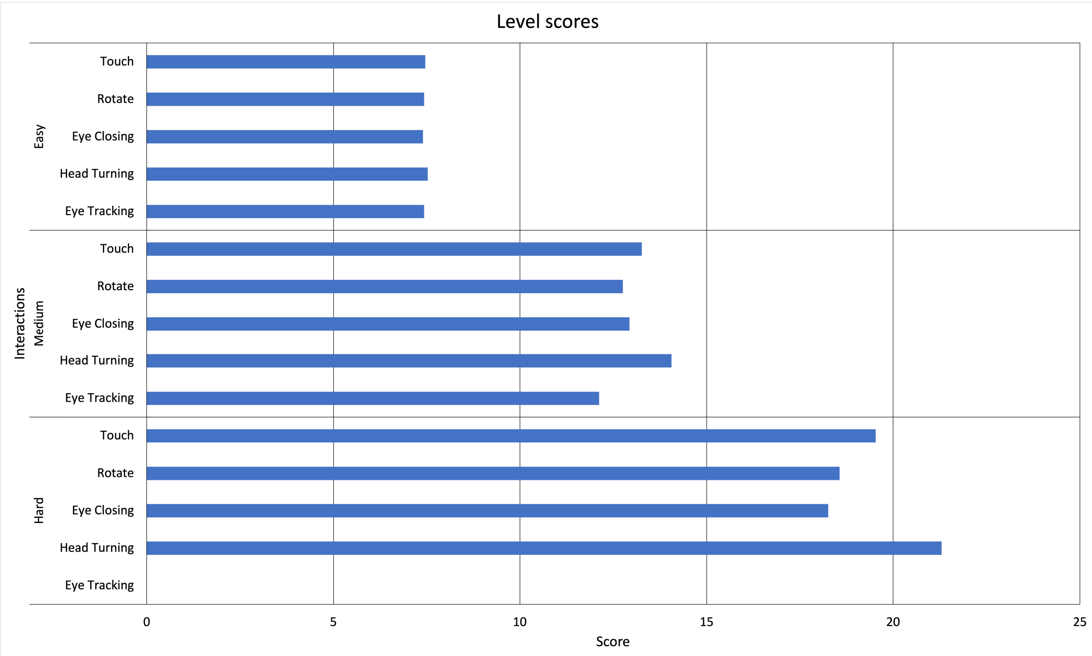

Figure 26. Figure © Ilkka Karjalainen, level scores

5.4. Level Completion Time

Figure 27 shows the results about the level completion times that players could get when finishing the level successfully. Microsoft Excel's Single Factor ANOVA was used to the data and there were no statistically significant results because all p-values were $p > 0.05$ (Appendix 1 : Table 13). The p-value is used to determine are the results happened by a chance. The p values were in level Easy $p = 0.623$, level Medium $p = 0.222$ and level Hard $p = 0.382$ (Appendix 1 : Table 13).

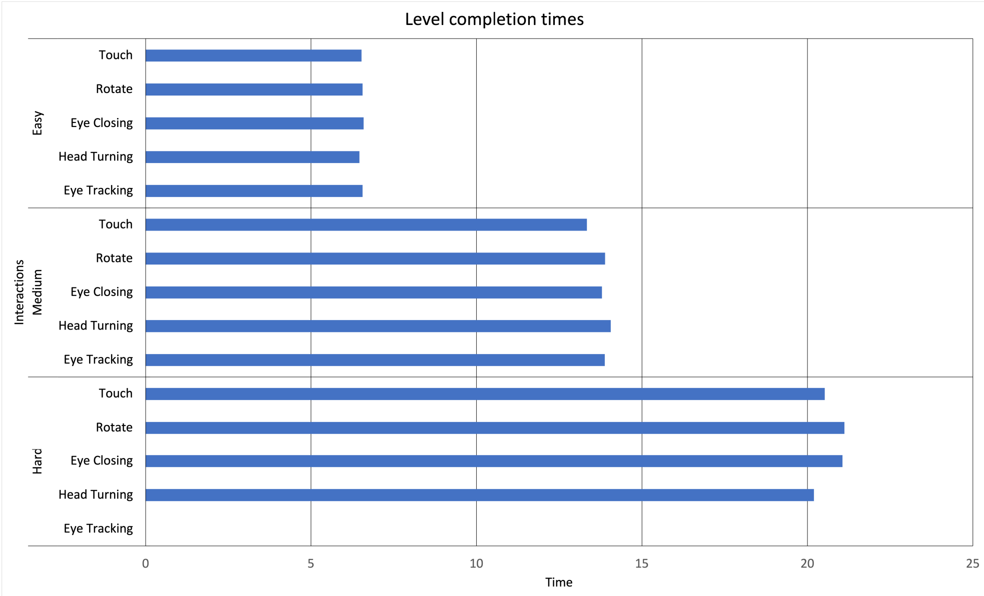

Figure 27. Figure © Ilkka Karjalainen, level completion times

5.5. Circle Movement

The circle movement tracks how much the Player has rotated the circle while completing the level successfully. The circle movement is measured as throughput, where throughput is the rate at which rotating the circle is processed. The throughput is calculated by dividing the rotated angle with rotating time, giving how many angles are rotated per second. Angles per second tell how fast the circle is rotated incorrectly, so the lower value is favorable. Also, the rotation count is logged, and the rotated angle is divided by the rotation count. That gives angles per turn which tell how many angles were rotated in one turn. The higher value indicates that more work is needed to rotate the circle in the correct position, so a lower value is favorable. Circle movement is separated into sections, angles per second and angles per turn.

5.5.1. Angles per Seconds

Figure 28 shows the results of the angles per seconds results in clockwise and Figure 29 counterclockwise that players were able to get when finishing the level successfully. Microsoft Excel's Single Factor ANOVA was used to the data, with statistically significant results in all the levels because they p values all lower than 0.05 (Appendix 1 : Table 14 and Table 15). The p-value is used to determine are the results happened by a chance. The results from measurements are presented in Table 4 and Table 5 which summarizes Tables Appendix 1 Table 14 and Table 15.

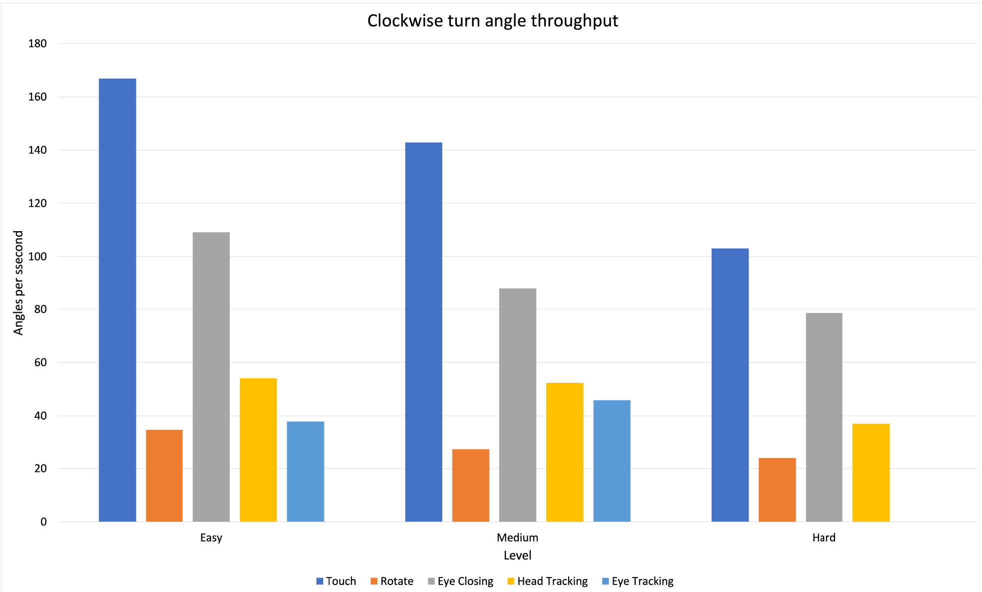

Figure 28. Figure © Ilkka Karjalainen, throughput of clockwise turn angle

Table 4. Clockwise angles per second results

Level	Touch	Rotate	Eye Closing	Head Tracking	Eye Tracking	p-value
Easy	166.83	34.77	109.10	54.11	37.90	$p < 0.05$
Medium	142.87	27.24	88.06	52.39	45.88	$p < 0.05$
Hard	103.06	23.93	78.59	37.08	0	$p < 0.05$

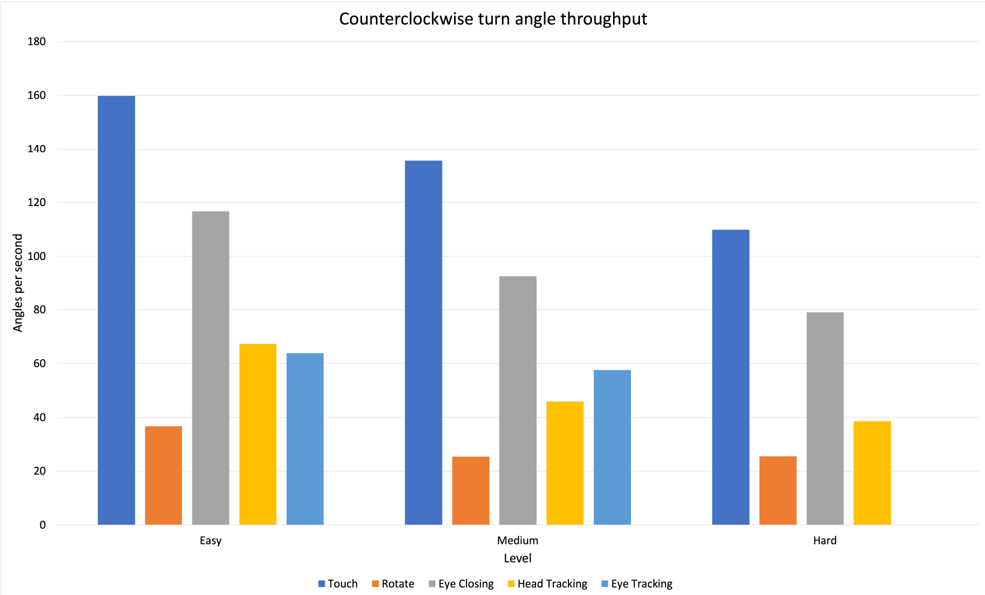

Figure 29. Figure © Ilkka Karjalainen, throughput of counterclockwise turn angle

Table 5. Counterclockwise angles per second results

Level	Touch	Rotate	Eye Closing	Head Tracking	Eye Tracking	p-value
Easy	159.82	36.72	116.70	67.26	63.82	$p < 0.05$
Medium	135.48	25.36	92.65	45.87	57.53	$p < 0.05$
Hard	109.84	25.44	78.97	38.58	0	$p < 0.05$

5.5.2. Circle Rotation

Figure 30 shows the results about the angles per turn results in clockwise and figure 31 counterclockwise that players were able to get when finishing the level successfully. Microsoft Excel's Single Factor ANOVA was used to process data statistically. All the results have a p-value of more than 5% from results, so there is a good chance the results are not a fluke and therefore statistically significant. Nobody managed to finish level Hard with Eye Tracking, and therefore there is no result from it. The results from measurements are presented in Table 6 and 7 which summarizes Tables Appendix 1 16 and 17.

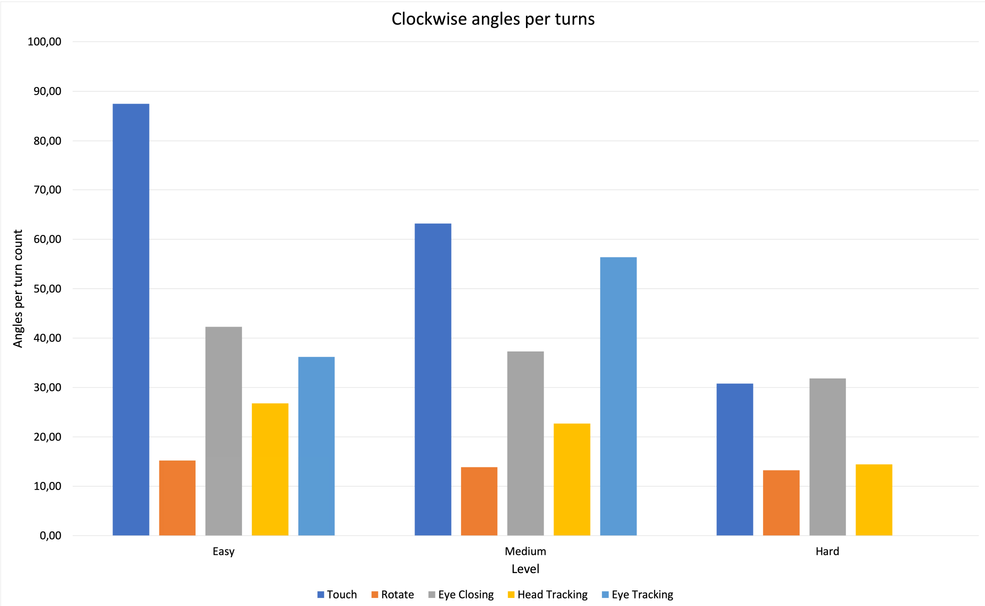

Figure 30. Figure © Ilkka Karjalainen, clockwise angles per turns

Table 6. Clockwise angles per turn results

Level	Touch	Rotate	Eye Closing	Head Tracking	Eye Tracking	p-value
Easy	87.49	15.21	42.29	26.80	36.18	$p < 0.05$
Medium	63.17	13.85	30.96	22.71	56.38	$p < 0.05$
Hard	30.81	13.21	1.10	14.42	0	$p < 0.05$

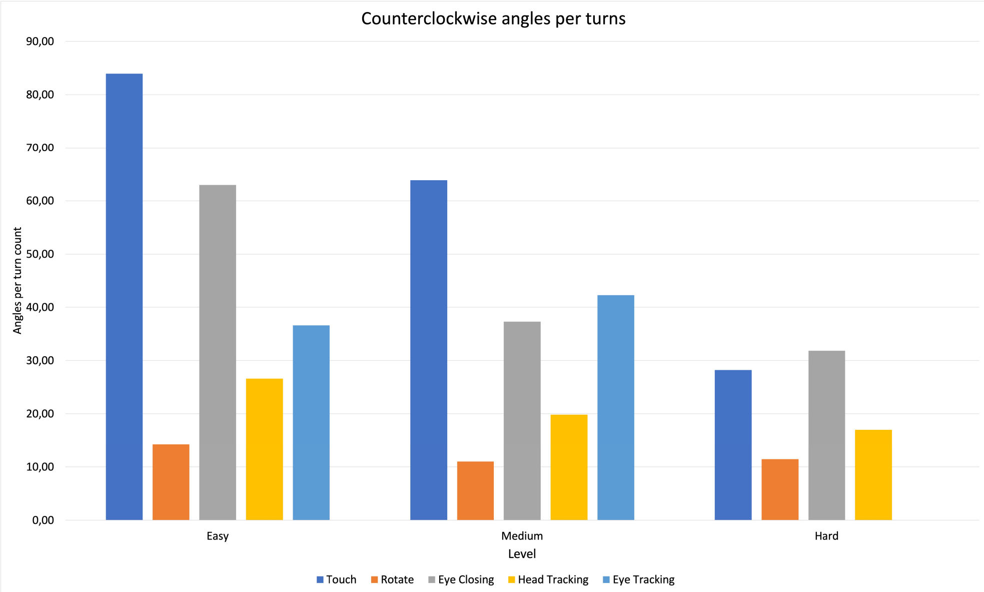

Figure 31. Figure © Ilkka Karjalainen, counterclockwise angles per turns

Table 7. Counterclockwise angles per turn results

Level	Touch	Rotate	Eye Closing	Head Tracking	Eye Tracking	p-value
Easy	83.94	14.25	62.95	26.61	36.60	$p < 0.05$
Medium	63.84	10.98	37.30	19.82	42.31	$p < 0.05$
Hard	28.22	11.44	31.82	16.98	0	$p < 0.05$

5.6. Failure Arc Count

From the arcs left count Appendix 1 : Table 20 is shown that even though nobody managed to finish level Hard with Eye Tracking, there was at least one case where only one arc was left when the level ball got out of the game area. Also, if arcs left min column is observed, it is shown that there were games played with every interaction in each level that game was almost finished successfully. When comparing the arcs left average and arcs left max columns, the Eye Tracking has the most arcs left when the game was finished unsuccessfully. That confirms that Eye Tracking is the most challenging interaction method. The Head Tracking has the seconds most arcs left, so it is the fourths hard interaction method that successfully finishes the level. Microsoft Excel's Single Factor ANOVA was used to the data, with statistically significant results in levels Medium and Hard with $p < 0.05$ (Appendix 1 : Table 19. The results are not statistically valid with level Easy because in level Easy $p = 0.129$ (Appendix 1 : Table 13). The results on average are presented in Appendix 1 : Table 19.

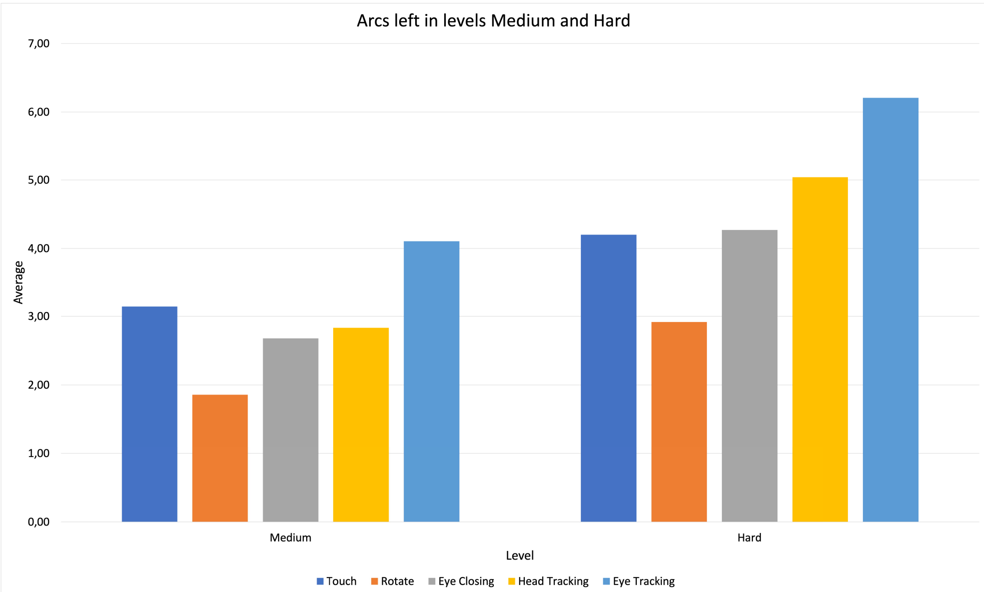

Figure 32. Figure © Ilkka Karjalainen, arcs left in levels Medium and Hard

Table 8. Failure arc count

Level	Touch	Rotate	Eye Closing	Head Tracking	Eye Tracking	p-value
Medium	3.14	1.86	2.68	2.83	4.10	$p < 0.05$
Hard	4.20	2.92	4.27	5.04	6.21	$p < 0.05$

5.7. Filtered Movement

The filtered movement collects information on how much movement needs to be filtered out to make the game more playable. Like in Touch interaction, finger dragging can shake slightly, and this shaking can be interpreted as changing direction, so this needs to be filtered out. In Rotation, the accelerometer reports a slightly different angle multiple times in a second, so this wobbling is also filtered. The Eye Closing is not filtered because eye closing is detected as an on/off method. Head Tracking and Eye Tracking also have this shaking, in Eye Tracking more than in Head Tracking. In figure 33 is collected Microsoft Excel's Single Factor ANOVA statistically processed data. All the results have a p-value more than 5% from results, so is results are not happened by chance and therefore are statistically significant (Easy $p < 0.05$, Medium $p < 0.05$ and Hard $p < 0.05$, Appendix 1 : Table 18). The filtered count in average is presented in Appendix 1 : Table 18.

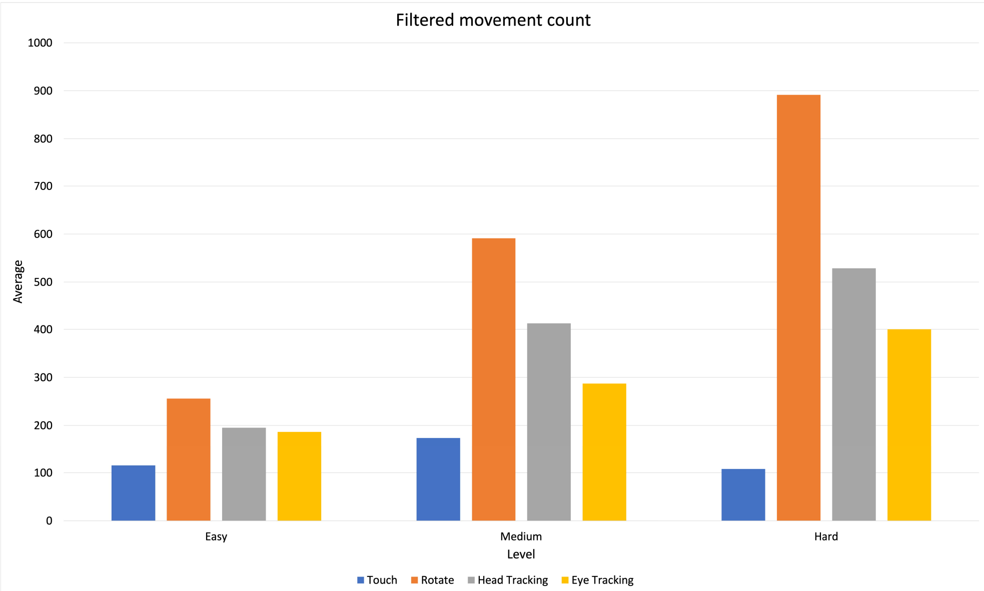

Figure 33. Figure © Ilkka Karjalainen, filtered movement count

Table 9. Filtered movement count

Level	Touch	Rotate	Head Tracking	Eye Tracking	p-value
Easy	115	256	195	187	$p < 0.05$
Medium	174	591	413	287	$p < 0.05$
Hard	108	892	529	400	$p < 0.05$

5.8. End Level Questionnaire

After the Player has completed the level with certain interaction, a questionnaire about the experience is presented where feedback on how easy it was to play the game, gaming interface, etc., with selected interaction method. Since eyes were used to control the game, a few questions about eyes were also asked. Questionnaire responses were entered with a slider on a 7-point Likert scale, where seven strongly agree, and one strongly disagrees with a statement. There were 17 Players where each other answered 13 questions which were:

1. Overall, I am satisfied with how easy it is to play this game
2. It was simple to play this game
3. I could effectively complete the objectives and challenges
4. I was able to complete objectives and challenges quickly
5. I was able to complete objectives and challenges efficiently
6. I felt comfortable using this system
7. It was easy to learn to play this game
8. Whenever I make a mistake in the game, I recover easily and quickly
9. The organization of information on the game screens is clear
10. The interface of this game is pleasant
11. I like using the interface of this game
12. While playing the game, my eyes felt excessively watery
13. While playing the game, my eyes felt dry

The first 11 questions are about the game interface, playing challenges, and playing in general and taken from study Barnett et al. [89]. The last two questions were about the does the playing with eyes cause issues with eyes. The results from first 11 questions with each level is presented in figures 34, 35 and 36. Results from the questionnaire were processed with Microsoft Excel's Single Factor ANOVA. Results from questions one to eight have a p-value lower than 5% from results, so results are not happened by chance and, therefore, statistically significant. Question nine had p-values higher than 5% from the results, so their results are not statistically significant. The last questions, twelve and thirteen, are taken from the convergence insufficiency symptom survey (CISS). The results had a p-value of more than 5%, so the results can't be differentiated from chance, so the results are not valid.

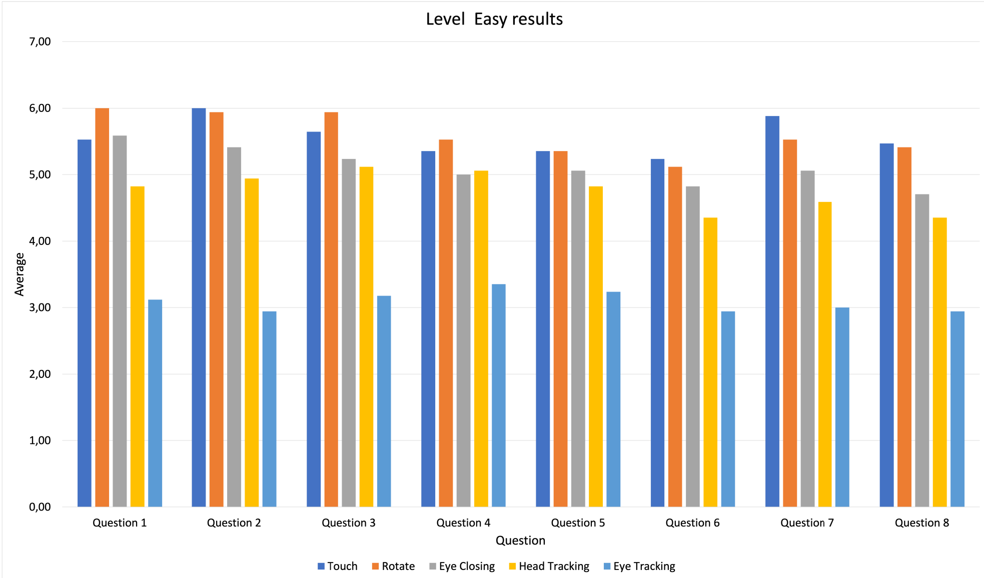

Figure 34. Figure © Ilkka Karjalainen, level easy results

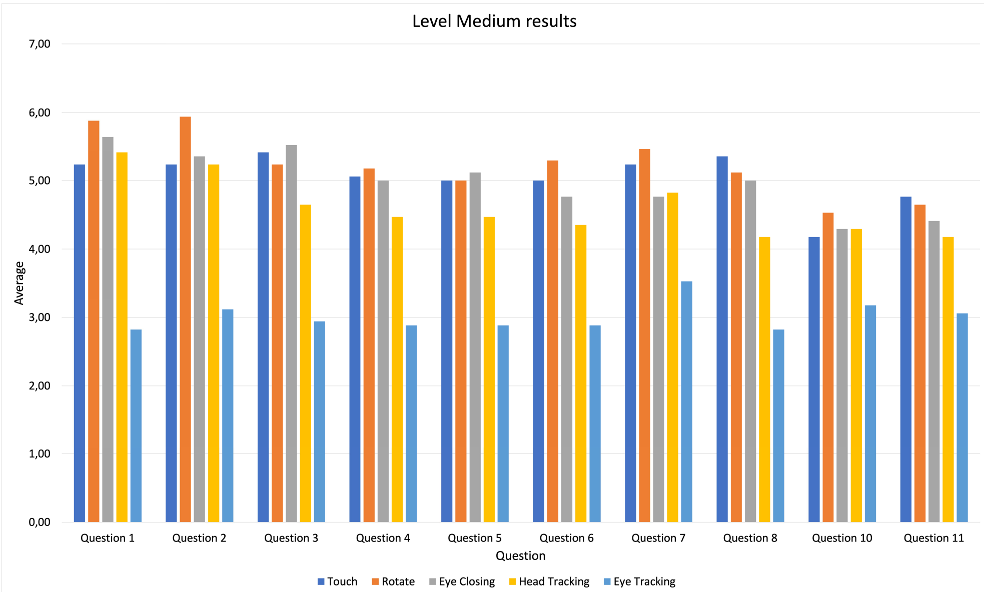

Figure 35. Figure © Ilkka Karjalainen, level medium results

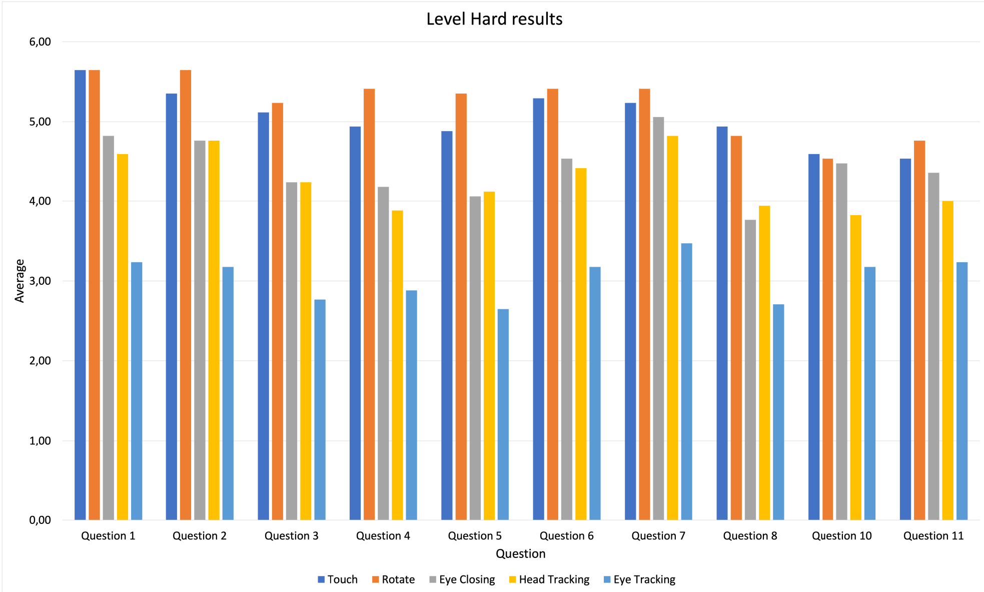

Figure 36. Figure © Ilkka Karjalainen, level hard results

5.9. Other Postquestionnaire User Feedback

Players were given the opportunity to give feedback from the testing session. Four themes were recognized from the material: Eye Tracking Difficulty: It's tough to navigate with your eyes. They tend to do things so quickly. Eye Tracking difficulty results were similar in study by Abbaszadegan et al. [6] where the eye-tracking interface was fatiguing and hard to use. New ways to play: There are many different interaction methods. Interaction benefits: Eye-tracking, especially if it were something that can be beneficial for sports (where control of gaze direction and use of peripheral vision is essential). Fun experiment: Eye-tracking game mode was hard. But in general, it was fun to try those different tracking methods.

With the questionnaire, players were able to give personal opinions about the DippaOut. According to Figure 24 the Touch and Rotate were the most favorable interaction methods, but Eye Closing was quite popular also. The Eye Tracking was the least suitable interaction method. When asking about what interaction-based games tested would like to see in the future, the Touch, Rotate, and Eye Closing was the most popular interaction methods. The players mainly have played games with touch interaction and a few with Rotation, but there were also a couple of players who said that they had played games with body tracking.

Open questionnaire items and other feedback went through a thematic analysis [95]. Despite the think-aloud protocol and open questionnaire items, there was not enough material to warrant a more thorough qualitative analysis. The purpose of collecting this material was to complement the quantitative results mainly by providing insight on what aspects during the study may have influences the results, especially if something peculiar is observed in the data analysis. The level completion status Figure 25 shows that Eye Tracking was the most difficult interaction method, and that can also be seen from the results thematic analysis because Eye Tracking issues pop up from the questionnaire. The recognized themes can be found from the following Table 10.

Table 10. Thematic analysis

Theme	Comment
Eye Tracking difficulty	"Using eye tracking is quite tricky."
New ways to play	" I never tried eye and head tracking before."
Interaction benefits	"Eye-tracking would be very beneficial for certain disabilities, that would be great. And head turning would be beneficial for physical health possibly, that would help me a lot. Good start and gave me ideas on how we could help people with these interactions. Lot of potential to grow."
Fun experiment	"It was fun to participate, All the experience was pleasant and the opportunity of trying new methods for playing and using phone was very interesting."

6. DISCUSSION

In this study, player experiences were observed and compared between different interaction methods using a DippaOut mobile game. Mixed data was collected during gameplay and with a questionnaire. Analyzed data revealed differences between interactions on gameplay difficulty and gameplay data and players' opinions.

6.1. Observed Difficulty

In observed difficulty, the goal is to determine the order of the interactions when comparing how successfully players could finish the levels, and measurements show clearly that players finished the levels easiest with Touch and Rotate. During testing, players were given the opportunity to try as many times to finish the level as they wanted and play the level again with the same interaction if they wished to. Figure 25 indicates that the Eye Tracking interaction method is the most challenging interaction method, which can be seen that it has the lowest success percent every level as seen in Appendix 1 : Table 11. The easiest interaction method is rotate, which has the highest success percent in every level shown in Appendix 1 : Table 11. The Rotate and Touch success percent are pretty even in the easiest level, but in the other levels, the difference between those two is more apparent. Head Tracking and Eye closing are in the third and fourth positions, which comes to difficulty. When comparing, Eye Closing has a better success percent in level Medium, but Head Tracking has a better percent in Easy and Hard. One reason for that could be that some players had issues with closing eyes; they could only close either left or right eye, not both. When looking at the play count column, it is shown that playing level Medium with Rotate is the least played. Also, its success percent is the fourth-best on total played percents and the best percent in all interactions in level Medium. Also is mention-able that level Hard with interaction Eye Closing is the most playable. It seems that players enjoyed playing with Eye Closing that they wanted to finish level Hard and kept trying several times before giving in. With Eye-tracking and Head tracking, players give in sooner.

In level Easy, level Medium, and level Hard, the p-values were too high, so scores were so close to each other that they were not statistically significant as can seen from Appendix 1 : Table 12. More players would have been needed to conduct more tests to get more statistically valid results. And in DippaOut the game design is so that only variable what changes is the interaction method, the ball speed and arc count stays the same.

The results of level scores are shown in Appendix 1 : Table 12. The scores are similar within all interactions in each level, and when checking, the p-values of each level are too high, so the results are no statistically significant. The goal of observing the score was to determine if the players aim for the top score by trying to break arcs with the same color to execute combos. With combos, the score would have been higher. Still, as observing the gameplay during test sessions, the players did not care about the combos and were only aiming to finish the level successfully. And since the remaining time of the level is added to the score when finishing the level successfully, we can see from Appendix 1 : Table 13 the completion times are almost the same, so no differences with the score because of the completion time.

Appendix 1 : Table 13 presents results of times successfully finished levels, and from there can be seen that there are no statistically significant results in any level. Even though there are differences with circle rotation speed with interactions, it did not affect level completion times. Also, more players would have to get statistically significant results because nobody could finish the level Hard with Eye Tracking. And also, in level Hard, only six players were able to finish the level successfully but in Touch and Rotate, the success count was more than double. But also, when looking at the level completion times in level Easy, where more players were able to finish a level successfully, there were no statistically significant results. From Appendix 1 : Table 13 can be seen that Touch was the fastest interaction method in level Easy and Medium, but in level Hard, the Head Tracking was the fastest interaction, so no one interaction pop out from the results to be the fastest interaction method.

It was expected that level times would be quite the same in every interaction because the level and ball speed are the same. In raw data, there were in some level completion times which were clearly under the average. Those might be cases where the ball is hit two arcs at once, meaning that the game physics are simulated and not like in real life. The ball position is updated in every frame as the game runs. After updating the ball position, it is checked does that ball overlap any arcs, and in some rare corner cases, the ball will overlap two arcs at the same frame, and two hits are reported, and two arcs disappear at the same time. So even though there were some outlier in score and level completion times the players were amazed when they managed to remove two arcs with one hit, those anomalies were smoothed away when averaging the scores.

Even though nobody managed to finish the level Hard with Eye Tracking, from Appendix 1 : Table 20 can be seen that in some cases, there was only one arc left when failing the level. So with more players, there could be cases where somebody manages the finish level Hard with Eye Tracking.

6.2. Angles per Seconds

Rotate was the most precise interaction method because players could rotate the circle in the right position easily in both directions. Appendix 1 : Tables 14 and 15 shows a comparison of throughput, angles per seconds, during playing level when level was finished successfully. With angles per second can be seen that how the angles are processed and what faster is the throughput that faster players were able to rotate the circle in the correct position. Eye Tracking has the second's best average on angles per second, although nobody could finish the level Hard with it. Head Tracking is in the third position. The Eye Closing, even though some players had issues with closing either of the eyes, was in the fourth position. The Touch was in the last position.

6.3. Circle Rotation

Results from Appendix 1 Tables 15 and 16 can be seen that Rotate is the most precise interaction method when inspecting how much in angles per turn players have rotate the circle. The Rotate has the lowest angles per turn average than other interactions, so players have managed to rotate the correct circle position at once. The second best interaction is not easy to say because Eye Closing and Head Tracking have both one-second positions, and Eye Tracking has no value in level Hard. Eye Closing is in the fourth position, and Touch in the last position.

6.4. Filtered Movement

From looking at the Appendix 1 Table 18 is easy to make a conclusion that Rotate interaction has the most filtered out movements. In filtered movement, the Eye Closing interaction is not included because it was the only interaction interpreted as an on/off interaction method. It is not a big surprise since the phone's gyroscope reports a new position even when the phone is left rest on a table. Where Rotate has the most filtered movements, the Touch has the least. When testing, the players could rotate the circle in the right position quite easily, and there was not so much extra shaking with it. Eye Tracking is the second place which is quite surprising since it was the most difficult controlling method. Eye-tracking position in second place can be explained by the players who were able to finish levels with Eye Tracking focused eyes so precisely and were even not breathing to minimize eye movements. Head tracking is in the third position, but players liked the Head Tracking so much that extra movements did not disturb the playing experience.

6.5. Failure Arc Count

The Rotate was the best interaction, and Eye Tracking the worst interaction according to the failure arc count. In failure arc count is inspected how close players were to finish a level successfully, and the Rotate interaction has the fewest arcs left on average when finishing level unsuccessfully. In level Easy, the results are no statistically valid, but in Medium and Hard, they are, as can be seen in Appendix 1 Table 19. The players were the closest to finish the levels with Rotate interaction. There were less than two arcs left on average from 8 arcs in level Medium, and in level Hard less than three arc on average from 12 arcs. So players managed to play $\frac{3}{4}$ of a level before failing. The interactions Touch and Eye Closing are in second and third position with about one arc more in level Medium and Hard. Appendix 1 Table 19 shows that Head Tracking is in the fourth position and Eye Tracking in the last place.

6.6. Interaction Order by Measurements

From measurements can conclude that Rotate was the best interaction to play with. It has the overall best level completion percent, which can be seen from Appendix 1 : Table 11. It has the the lowest throughput Appendix 1 : Tables 14 and 15 so it took the least amount of rotations from players to get the circle in right position. The Rotate also has the smallest number of turned angles when rotating the circle. And Rotate also has the least arcs left when the level was finished unsuccessfully as shown in Appendix 1 : Table 19. The Rotate has the most filtered movements as seen in Appendix 1 : Table 18 but since filtered movements do not affect gameplay, it is just informative metric importance is less than the other measurements. And since there were no statistically valid data in level completion times which are shown in Appendix 1 : Table 13 nor score from 12, they have not been researched to give any value for order of the interactions. So because of the good points of the Rotate interaction game, designers should create more games specially designed to be played with rotating the phone.

The Rotate was the best interaction derived from the measurements, and the worst interaction was the Eye Tracking because it has the worst level completion as percent as seen in Appendix 1 : Table 11. It also has the most arcs left when failing the level as shown in Appendix 1 :Table 19. It is in second place in filtered movements, but this metrics is not so important as mentioned before. And after all, nobody managed to finish the level Hard with Eye Tracking; comparing it to the other interactions is hard with measurements because it lacks the level Hard results. So level completion metric is used to judge Eye Tracking to the worst interaction.

Where the best and the worst interaction were easy to see from the results, the order of remaining interactions, Touch, Eye Closing and Head Tracking was not so easy to see. The Touch has cleary the best level completion success percent when comparing to these other two interactions from (Appendix 1 : Table 11) the players really like to play with Eye Closing as seen in results. And since gameplay count was not limited to certain amount of tries the amount of games played with Eye Closing biases the results. And even thought the success percent was good with the Touch interaction Appendix 1 : Tables 14 and 15 show that players had to rotate the circle more with touch than any other interaction. So, because of the mixed results where these three interactions position within different metrics there is hard to say which interaction is the better than another. To make the difference, some metrics are dropped from the analysis and the only metric that is used is the most important one, the level completion results which tells how easily the player were able to finish the levels. From Appendix 1 : Table 11 can be seen that Touch has the highest percentage from these three interactions, Head Tracking has the second-highest, and Eye Closing the third highest.

6.7. Other Gameplay Experiences

The postquestionnaire provided information about players' opinions which are compared to measurement data and analyzed. The questions are divided into sections where questions 1 and 2 are about easiness and simpleness of interaction method when playing. Questions 3 to 5 how able players were to complete levels with each interaction method. Questions 6 to 8 are how comfortable the players found the interaction to be when learning to play the game and how easy it is to recover from mistakes with different interactions. Question 9 focuses on how the players could be on track on their level of progress when playing. Questions 10 and 11 are about how they liked to play the game with each interaction. The last questions, 12 and 13, are mainly targeted to the interaction where eyes were used to determine if there any issues that would interfere with playing the game with eyes.

6.7.1. How Easy Is the Game to Play?

From players' answers to the first question from Appendix 2 : Table 21 can be seen that players liked the Rotate interaction most which can be explained when comparing their results to level completion status results as seen in Figure 25. The Rotate has the best success count to finish levels successfully. So players liked the Rotation interaction the most because they could finish levels most efficiently with that. Also, because DippaOut is about rotating the game area is understandable why players liked to play it by turning the mobile phone. From answering this question, the players' opinion was that Eye Closing was the seconds most satisfying interaction to play with. Even though the players had issues finishing levels successfully as seen in Figure 25 they kept trying, especially in level Hard, they tried to finish more than in other interactions. And also, from Appendix 1 : Table 19 players were close to finishing the levels with Eye Closing interaction; there can be a possibility they found the Eye Closing interaction method quite intuitive. Eye Closing also points out that players might find these new ways to play enjoyable when designing games and interactions when thinking outside the box and want to play more games with different interactions. The Touch is in the third position, and also, it was the interaction with the most successfully finished levels. So even though players were used to playing with touching the screen, rotating the phone seems to be easier, and eye closing might be better than touching because it is a new way to play. Head Tracking is the fourth position even it has a better success rate to finish levels than with Eye Closing, but as seen from Appendix 1 : Table 19 when failing the level, there were more arcs left than in Touch, Rotate, and Eye Closing. Eye Tracking was in the last position, which is no a big surprise because it was the hardest interaction method to finish the levels as seen in Figure 25. The second question relates to previous questions because it asks the player how easy it was for each interaction method. So the results are also similar. Players opinion about game playing simplicity is that the Rotation is the simplest interaction to play the game as seen from Appendix 2 : Table 22 which seems to be consistent with results from Figure 25 and Figure 19. Rotation is a simple interaction method to finish the level successfully, and when the level is failed there, the least arcs left. The Touch is the second simplest way to play the game, and Eye Closing the third. These two interaction methods are reversed from section 6.7.1 question. Still, the differences are quite marginal in level Easy and Medium as seen in Appendix 2 : Table 22 and level Hard about a half-point. The Head Tracking and Eye Tracking are in the same order as in section 6.7.1 with the same reasons.

6.7.2. Gameplay Effectiveness

When inspecting the results, how effectively players accomplished completing the levels from Appendix 2 : Table 23 on their own opinion, the result is again Rotate which can be backed up with results from Figure 25. The Eye Tracking is judged to be the worst way to complete the level successfully, which can also be verified with Figure 25. Eye Tracking is the worst interaction method when comparing effectiveness because of the game design, and players follow the ball with eyes. Still, instead, they should be using eyes to rotate the circle in the correct position.

So players' opinions match the data carried from testing. The Touch is in the second position, which matches the collected data, Eye Closing on third and Head Tracking in fourth. Also, in Eye Tracking, some players said that they could only close either left or right eye, not both, which might explain why Touch is judged to better than Eye Closing and players were more familiar with Touch than Eye Tracking as shown in Figure 20. Where the previous section question focuses ineffectively, this question focuses on quickness. How quick players thought that they were able to finish a level successfully with each interaction. Appendix 2 : Table 24 shows in every level; the Rotate is the quickest interaction method to complete the level successfully. This differs from measurement data because Rotate was the third-fastest interaction method, as seen in Appendix 1 : Table 13. Players might have thought that because they managed to finish levels efficiently with Rotate interaction, they have completed the levels quickly also. The Touch was the second quickest interaction method in the player's opinion, and Appendix 1 : Table 13 shows that in this case, players' opinions match the measurement data. In the players' own opinion, the Eye Closing was the third quickest interaction method, and Head Tracking the fourth. Players' view is that Eye Tracking the slowest interaction method, but when comparing level completion times in Appendix 1 : Table 13 that is the fastest interaction method. Even though the data in Appendix 1 : Table 13 is statistically valid only in level Easy, the Eye Tracking is the fastest in that level too. The players might found Eye Tracking so hard that they did not realize that they managed to finish the level quickly with it. The Eye Tracking was also the most challenging interaction method according to the Figure 25 so it had to least amount of successfully finished levels which might lower the average completion time if only the most skilled players were able to finish levels successfully. The only objective in DippaOut is to finish the level successfully. There are also secondary objectives to get a high score and finish the level quickly, but when observing players playing sessions, the focus was only to complete the level successfully; the players did not care about the time or score level. In some cases, players wanted to play the level again with the same interaction just for fun or due that the interaction was something new that they have not used before (Eye Closing, Head Tracking). The players felt that Rotate interaction was the best interaction to finish levels successfully. The second best interaction to finish levels was Touch which might be because it was the most familiar interaction to players as seen in Figure 20. The third place of the interaction in completing the levels is Eye Closing in players opinion as described in Appendix 2 : Table 25 but according to the data in Figure 25 players were able to finish the levels Easy and Hard better with Head Tracking than Eye Closing. By players' answers, Head Tracking is the fourth-best interaction method to complete the levels and Eye Tracking the worst.

6.7.3. Interaction Suitability for Playing

The DippaOut was designed to play with rotating the circle, and according to Appendix 2 : Table 26 and Figure 25 players were the most comfortable playing the game by rotating the phone. With a different kind of game design, the results could be different. When looking at data collected during testing sessions, the Rotate interaction has the least arcs left when failing the level as shown in Appendix 1 : Table 19, so this backs up that players were comfortable using Rotate interaction. In Rotate, there were a lot of filtered movements from Appendix 1 : Table 18, but that did not interfere with playing notably, so players' handshaking were filtered out. Also when inspecting Rotate throughput Appendix 1 : Tables 4 and 5 is has the lowest throughput so with it the players were able to rotate the circle in right position fastest. The Touch felt the second-best interaction to play DippaOut, which lines with Appendix 1 : Table 19 results where the Touch is shared second place when measuring how many arcs were left when failing the level. When inspecting the level completion results from Figure 25 the Touch is in the second position. Even though Touch took the biggest turn angles Appendix 1 : Tables 6 and 7 in did not bother players gameplay experience. Eye Closing is in the third position when measuring comfortableness which can be seen in measured data because Eye Closing interaction positions mainly in the third position. From Figure 25 shows that even though Eye Closing was not the easiest nor the hardest interaction method, players liked it because they tried so many times to finish the level Hard successfully. Head Tracking was judged to be in the fourth position on comfortableness which is consistent with measurements as seen from Appendix 1 : Table 25. Eye Tracking was judged to be in the last position, which is not a surprise because it was the hardest interaction according to the measured data and on players' opinions when rating the interaction in Figure 24. When estimating how easy it is to learn to play the game with each interaction, the most used interaction, Touch, and rotate were on the top positions. The DippaOut is played by rotating the circle, so players were quickly adapted to play the game by rotating the phone. And modern smartphones are used with Touch, so players are familiar with it. The easiest interaction to learn to play the game is to Rotate according to the questionnaire and be backed up with measurements, e.g., level completion data from Figure 25 and data from failure arc count data from Appendix 1 : Table 19. With those tables, it is easy to see why players chose Touch as the second-best interaction on easiness to learn to play DippaOut. The Eye Closing was a completely new interaction method for players. Still, players' opinion was that it is on the third position when comparing interaction easiness to learn to play the game. The Head Tracking was also a new interaction method for players, and they managed to finish level Hard better with it than Eye Closing players decided that it belongs to the fourth position. The most challenging interaction, Eye Tracking, was in the last place. Rotate was, in players' opinion, the most comfortable, the simplest, and efficient interaction method. The Touch was the best to recover from mistakes. When looking at Appendix 1 : Table 18 is it evident that the Touch has the least amount of filtered movements which indicates that there was not so much hesitation when playing with Touch interaction. The Rotation is the second-best interaction method to recover from mistakes which is understandable because it has the least amount of arcs left when failing, as seen in Appendix 1 : Table 19. The third and fourth position goes to Eye Closing and Head Tracking with only one point margin to each other. The most challenging interaction to recover from mistakes is Eye Tracking which seems to follow the trend that Eye Tracking is the most difficult interaction to control the game. In player's opinions, Rotate was the easiest interaction to learn to play the game according to Appendix 2 : Table 27. The Touch was in second place, Eye Closing the third, Head Tracking the fourth, and Eye Tracking was the last. Since players were familiar with using mobile phones with Touch as seen in Figure 20 so it is understandable that players chose the Touch to be the easiest interaction to learn to play the game. The Touch was the first interaction that players played each level which could have an effect on learning because, after each interaction, the level is more familiar. But in the players' opinion, the first interaction was the easiest interaction to learn to play. The order of interaction did not have an effect. When player made a mistake the best interaction to recover from it was the Touch according the players as seen in Appendix 2 : Table 28 even though it has more than $\frac{1}{3}$ arcs left when failing the levels as seen in Figure 32. The second on mistake recovery was Rotate, which continues the trend from level completion results from Figure 25. The Eye Closing in third position is a little bit better than Head Tracking on fourth position. In the end is the Eye Tracking in last position, which match the results from Figure 25.

6.7.4. Game Screen Information

There is plenty of in-game information on screen during gameplay: Score, time, angle, etc. Also, when using a camera, the distance between the player and phone is presented with color codes to guide players to keep the phone far enough from the eyes. When evaluating answers from players about the organization of information on the screen from Appendix 2 : Table 29 it is clear that there are no statically valid results. The players did not care about the time nor score; they aimed to remove all arcs and finish the level. The players did not replay the levels to get a better score, and they replayed for the sake of fun. This was not asked from the players but was mentioned during testing sessions. One point which also shows that players were not paying attention to information on the screen or the information was not accessible is that during tests with eyes, players were keeping the phone too close to their faces. There was red text on the screen indicating too close proximity of the phone during gameplay. Still, players did not notice that and were guided test observers that increasing distance between eyes and phone would increase the chance of finishing the level successfully.

6.7.5. Is Interaction Fun to Use

for the worst position, which is the Eye Tracking as it is the most difficult interaction method to finish the level as shown in Figure 25. Appendix 2 : Table 30 shows that when looking at the averages, there are not so many differences between interactions except the Eye Tracking, and their p-values are too high to separate results from a chance in level Easy, but in levels, Medium and Hard results are statistically valid. Since Rotate and Touch were the easiest interaction methods to finish a level successfully, it is easy to see why the players ranked both interactions to the first position on interaction funniness. The Eye Closing and Head Tracking were quite even on players' opinions, but the Eye Closing was a little bit better. When asked did the players like using the interface, the results are shown in Appendix 2 : Table 31 which shows that the p-value is too high on level Easy but levels Medium and Hard have statistically valid results. From the results, it is clear that Eye Tracking is not fun to use; its average is clearly under neutral value (four), where the other interactions have an average neutral or more. Since the Eye Closing was the hardest interaction to finish level with as seen in Figure 25 is understandable that players found that Eye

Tracking is not fun to use. The second least fun interaction to use is Head Tracking, which can be backed up with the results from Figure 25. The Eye Closing is in the middle of interaction ranking order in the third position. The first position of the most fun to use interaction is shared between Touch and Rotate.

6.7.6. Eye Fatigue During Gameplay

Questions from sections 9.12 and 9.13 are investigated only with interaction Eye Tracking and Eye Closing because these questions are focused on eyes. From Appendix 2 : Tables 32 and 33 the p-values are too high so these questions had no statistically significant results on this aspect of player experience. There is no clear reason for why results were not statistically valid, but it might be a sum of several things:

- Nor the Eye Tracking or Eye Closing did not cause any inconvenience to eyes. This can also explain why players tried the Hard level with Eye Closing so many times. If it had hurt their eyes, they would not have tried so much.
- The Eye Tracking was such a hard interaction method that players did not play so long with it and when failing Easy or Medium tried only once with Hard.
- The Eye Tracking provided a good challenge and was fun to play with it.

6.7.7. Results of a Question about Gameplay

To summarize results questions about game playing with different interactions a graphs for each level is presented in Figure 34, Figure 35 and Figure 36. When looking at those graphs, it is clear that Eye Tracking has the least amount of points, and its line is way below other lines. The Eye Tracking is not suitable interaction for playing a game like DippaOut, according to the players. There can be several reasons for that ranging from poor implementation to game design, but from the player, the questionnaire can also be seen that players were interested in it. The lines of Eye Closing and Head Tracking are pretty near each other, where Eye Closing is marginally better. These interaction methods were new to players, so their performance was not expected as good as more familiar interaction methods. The margin between the four remaining interaction methods is relatively small. In level Hard, the differences are more visible than in levels Easy and Medium. And like mentioned before, players liked playing with Eye Closing, which might explain why Eye Closing is in third position and Head Tracking in the fourth position. The order of the remaining two interactions, Touch and Rotate, is found out when inspecting their performance at every level. In levels Easy and Medium, the points are distributed relatively evenly with both of the interactions, but in Level Hard, the difference is much more significant in favor of Rotate. From the players' opinions, a conclusion is derived that Rotate is the most suitable interaction for DippaOut. When asking to rate the interactions at the end of the questionnaire, Touch's interaction was most ideal in the player's opinion. Still, when examining the results of the end level questionnaire, the Rotate was better.

6.8. Limitations and Outlier Observations

During gameplay sessions, some problems were identified, and when talking with players, some observations were made.

6.8.1. Problem with DippaOut and Eye Tracking

Some interaction methods were more suitable for playing the DippaOut than others; one comment from several players pointed out Eye tracking. First of all, precise controlling with eyes is challenging because commanding eyes is not so simple, like the blinking of eyes can cause unwanted circle rotation. The second problem is that to make eye controlling work, there is a need to move the phone further from the players' face, and it needs to be played without eyeglasses; some of the players said that they could not see the yellow ball on-screen, which made the Eye-tracking unplayable from them. With taking the phone closer, they managed to play, but they did not work to clear the level. The was also a flaw in the game design, which was pointed out by several players; they followed the yellow ball movement, and by doing that, they rotated the circle, which caused failures. They had glimpsed where that ball is going and then focus on turning the circle and follow the ball with peripheral vision. This made controlling quite vague and frustrating. There were also player eye position-related problems with the camera. If the player's eye sockets were deep and eyes small, the camera had issues detecting eyes and their direction. Since this was a hardware issue in the camera, it could not be fixed with the software, but this problem was fixed by pausing the game when eyes could not be detected and informing the player that eyes are not visible.

6.8.2. Problem with Not Limiting Playing Count

Because some interaction methods are easier than others, one should limit the number of times the player can try to clear the level. Some players played multiple times of the same level with the same interaction method because they wanted to finish the level successfully. Thus, players played some interactions far more often than others, which could affect results.

6.8.3. Eye Hand Coordination

One of the questions to arise from the data is the relation of controlling the game with eye closing and player handiness. Data indicates that turning the circle with the right eye and left eye are not equals. Some of the players used only left or right eye to control the game. Because that relation was not thought of when doing the questionnaire, there was no question about player handiness. Also, is it possible that the player prefers to close either eye because it is easier?

6.8.4. DippaOut Manual

The DippaOut manual to conduct the test procedure was four pages long. In hindsight, the manual was too long and too detailed, and some of the players asked questions covered in the manual, but the player has forgotten some details of the testing procedure.

6.8.5. Problem Closing Only One Eye

When inspecting how much player has rotated the circle both clockwise and counterclockwise during playing when finishing the level successfully. First checking from the Appendix 1 : Tables 16 and 17 there is an anomaly in Eye Closing interaction because there are much fewer clockwise turns than counterclockwise. This is explained by that some players said that they could only close only one eye.

6.8.6. Eye Closing

According to the level success percent, the second to last interaction method was the Eye Closing players seemed to enjoy playing and tried to finish the levels several times. This was shown on results that level Hard was played eighty times with Eye Closing interaction. The DippaOut seems to have a good game design for controlling with Eye Closing. And also, players were surprised that the game could be played with just closing eyes, and the uniqueness of the interaction might keep them trying, and they didn't lose interest in playing.

6.8.7. Interaction Order

The players played all the levels with the same interaction order, which was Touch, Rotate, Eye Closing, Head-Turning, and Eye Tracking. The order was decided to teach the player the gameplay with familiar interaction methods before interaction methods that used the camera. This ensures that the player who knows how to play the game can concentrate on interactions instead of learning the game. This gives the player a better chance to finish levels successfully with unfamiliar interaction methods, and this could cause interference in results. Since the player has already successfully finished the level with other interactions, it gives the player confidence that the level can be completed and makes the interactions easier than if the interaction order would have been randomized.

6.9. Future Work

Controlling games with the mind has been a gaming console manufacturer's goal for several decades; even early as 1984, a mind controller device for Atari 2600 was prototyped. The device was Atari MindLink [96] which was presented in CES 1984 and later canceled with three games still in development. Currently, Elon Musk's company Neuralink is developing technology brain-computer interfaces (BCIs) where brains are connected physically to computers and use the mind to control it. These BCIs might lead to future different kinds of applications and games where everything is designed from another perspective. A game like DippaOut might be easy to implement with NeuraLink because playing is done only by rotating the circle clockwise or counterclockwise, so rotating in either direction could be easily identified. There is also off-the-shelf Electroencephalography (EEG) monitors, like Muse 2, which is used for a multi-sensor meditation device that provides real-time feedback. With EEG devices, determining does the player wants to rotate the circle in either direction with concentration might be easily distinguished, and playing DippaOut could be possible.

7. SUMMARY

Mobile phones are a ubiquitous and popular platform for gaming. They provide various possibilities for controlling gameplay; however, they lack the most common ones joystick and keyboard & mouse. In this study, I implemented a prototype called DippaOut with five different interaction methods: Touch, Rotate, Eye Closing, Head Tracking, and Eye Tracking. The DippaOut has three different levels: Easy, Medium, and Hard. These different interactions are compared with data collected automatically during gameplay sessions, an end level questionnaire after each level, and an open questionnaire where players could express their opinions after the test session. The DippaOut's design is a successor of the classic arcade game Breakout [21] where the goal is to clear the level from blocks with a ball and a paddle. The DippaOut changes the game dynamics by removing the paddle and changing the game area from static rectangle to rotatable circle, where the blocks are changed to circle arcs which are removed from a hit. Removing the paddle removes controlling from left to right to rotating circle either clockwise or counterclockwise, which enables different interaction methods. The metrics collected during gameplay focused on how suitable the interaction methods were on DippaOut by calculating how much the circle was turned in either direction during gameplay, how many times the circle was rotated, and how many circle arcs were left when failing the level. Also, information about the score, level completion times, and level completion status was collected. The end level questionnaire was used to get player opinions from each interaction method in each level.

The results from level completion status and failure arc count showed that the Rotate was the most suitable interaction method to play DippaOut and Eye Tracking the worst. The measurements from the rest of the interactions gave mixed results, so no difference could be made of their order, and more research is needed for that. Since the only changed attribute during testing sessions was the interaction method, the level completion times and score when finishing the level successfully yielded no statistically valid results since after statistical analysis, their p-values were higher than 0.05, which means that results might have been occurred by chance.

Also, the end level questionnaire had similar results to in-game measurements from the most suitable interaction (Rotate) and the worst interaction method (Eye Tracking). The order of the remaining interactions gave mixed results again. One reason for that was that Eye Closing was a new interaction method for players, and since it was fun to use, it might forgive some flawless in the interaction method.

The order of interaction from the open questionnaire after the testing sessions were: Touch was the most suitable, then Rotate, Head-Turning, Eye Closing, and Eye Tracking as the last position as seen in Figure 24. Since eyes were used in playing, fatigue was also studied to see does playing negatively affects players, but no statistically valid results were found. There were, however, some issues with eye-based interactions because some players could close only one eye, either left or right but not both. Other issues found out during test sessions were, e.g., the game count should be limited because there were cases that players wanted to finish the level that they played the same level with the same interaction multiple times. For future work, it would be interesting to see how brain-computer interfaces could be used for playing games.

8. REFERENCES

[1] Piccione P.A. (1980) In search of the meaning of senet.

[2] Kent S.L. (2001) The Ultimate History of Video Games: From Pong to Pokemon–The Story Behind the Craze That Touched Our Lives and Changed the World. Crown; Illustrated edition.

[3] Curt Vendel M.G. (2012) Atari Inc.: Business is Fun. Syzygy Press.

[4] Niehenke E. (2014) Wireless communications: Present and future: Introduction to focused issue articles. Microwave Magazine, IEEE 15, pp. 26–35.

[5] Pajitnov A, Pokhilko V and Pokhilko A. (1984). Tetris [Video game]. USSR: Pajitnov A, Pokhilko V and Pokhilko A.

[6] Mahdieh Abbaszadegan Sohrab Yaghoubi I.S.M. (2018) TrackMaze: A Comparison of Head-Tracking, Eye-Tracking, and Tilt as Input Methods for Mobile Games. Kurosu M. (eds) Human-Computer Interaction. Interaction Technologies. HCI 2018. Lecture Notes in Computer Science 10903. DOI: `https://doi.org/10.1007/978-3-319-91250-9_31`.

[7] Hassan M., Magee J.J. & MacKenzie I. (2020) Evaluating hands-on and hands-free input methods for a simple game. In: HCI.

[8] Cuaresma J. & MacKenzie I.S. (2014) A comparison between tilt-input and facial tracking as input methods for mobile games. In: 2014 IEEE Games Media Entertainment, pp. 1–7.

[9] Erdem Y., Ince I., Kusetogullari H. & Sharif M.H. (2015) Computer game controlled by eye movements. International Journal of Scientific Research in Information Systems and Engineering (IJSRISE) 1, pp. 97–102.

[10] Bang A.L., Krogh P., Ludvigsen M. & Markussen T. (2012) The role of hypothesis in constructive design research.

[11] Goldsmith R. (1947), Patent us2455992a, cathode-ray tube amusement device.

[12] Turing A.M. (1937) On Computable Numbers, with an Application to the Entscheidungsproblem. Proceedings of the London Mathematical Society Volume s2-42, Issue 1, 230–265 p.

[13] Wolf M.J.P. (2007) The Video Game Explosion: A History from PONG to Playstation and Beyond. Greenwood; Illustrated edition.

[14] Kates, J. (1950). Bertie the Brain [Video game]. CA: Kates, J. .

[15] Bennett, J.M, & Stuart-Williams R. (1951). Nimrod [Video game]. UK: Bennett, J.M, & Stuart-Williams R. .

[16] Higinbotham, W, & Dvorak, R. (1958). Tennis for Two [Video game]. US: Higinbotham, W, & Dvorak, R. .

[17] Graetz J.M. (1981) The origin of spacewar 7.

[18] Kent S. (2000) The First Quarter: A 25-Year History of Video Games. BWD Press. URL: https://books.google.fi/books?id=ny-CAAAAMAAJ.

[19] The Galaxy Game. URL: http://infolab.stanford.edu/pub/voy/museum/galaxy.html. Accessed 16.6.2020.

[20] Atari (1972). Pong [Video game]. NA: Atari.

[21] Bushnell, N and Bristow, Steve and Wozniak, Steve (1976). Breakout [Video game]. US: Bushnell, N and Bristow, Steve and Wozniak, Steve.

[22] Nishikado, Tomohiro (1978). Space Invaders [Video game]. US: Nishikado, Tomohiro.

[23] Nutting, Dave (1976). Sea Wolf [Video game]. US: Nutting, Dave.

[24] Magnavox Odyssey. URL: https://en.wikipedia.org/wiki/Magnavox_Odyssey. Accessed 16.6.2020.

[25] Wolf M.J.P. (2012) Before the Crash: Early Video Game History. Wayne State University Press.

[26] Activision (1982). Pitfall! [Video game]. NA: Activision.

[27] Nintendo R&D1 (1981). Donkey Kong [Video game]. JP: Nintendo R&D1.

[28] Williams Electronics (1981). Defender [Video game]. NA: Williams Electronics.

[29] Clark P. (1982), The play's the thing. URL: https://archive.org/stream/byte-magazine-1982-12/1982_12_BYTE_07-12_Game_Plan_1982#page/n7/mode/2up/. Accessed 5.7.2020.

[30] Atari, Inc. (1982). E.T. the Extra-Terrestrial [Video game]. NA: Atari, Inc.

[31] Namco. (1980). Pac-Man [Video game]. JP: Namco.

[32] Nintendo EAD (1985). Super Mario [Video game]. JP: Nintendo EAD.

[33] Nintendo EAD (1986). The Legend of Zelda [Video game]. JP: Nintendo EAD.

[34] Square Enix (1987). Final Fantasy [Video game]. JP: Square Enix.

[35] Konami (1987). Metal Gear [Video game]. JP: Konami.

[36] Sonic Team (1991). Sonic the Hedgehog [Video game]. JP: Sonic Team.

[37] Capcom (1996). Resident Evil [Video game]. NA: Capcom.

[38] Rare (1997). GoldenEye 007 [Video game]. NA: Rare.

[39] Nintendo EAD (1998). The Legend of Zelda: Ocarina of Time [Video game]. NA: Nintendo EAD.

[40] Nintendo EAD (1996). Super Mario 64 [Video game]. NA: Nintendo EAD.

[41] Bungie (2001). Halo: Combat Evolved [Video game]. NA: Bungie.

[42] Hitmaker (1999). Crazy Taxi [Video game]. NA: Hitmaker.

[43] Smilebit (2000). Jet Set Radio [Video game]. NA: Smilebit.

[44] Sega AM2 (1999). Shenmue [Video game]. NA: Sega AM2.

[45] McNeil S. (2019) Hey! Listen!: A journey through the golden era of video games. Headline.

[46] Montola M., Stenros J. & Waern A. (2009) Pervasive Games: Theory and Design (Morgan Kaufmann Game Design Books). CRC Press.

[47] id Software (1993). DOOM [Video game]. US: id Software.

[48] Networks L. (2020), State of online gaming 2020. URL: `https://www.limelight.com/resources/white-paper/state-of-online-gaming-2020/`. Accessed 13.4.2020.

[49] Alavesa P., Pakanen M., Kukka H., Pouke M. & Ojala T. (2017) Anarchy or order on the streets: Review based characterization of location based mobile games. In: Proceedings of the Annual Symposium on Computer-Human Interaction in Play, pp. 101–113.

[50] Lai Y.R. & Philip Hwang T.K. (2015) Virtual touchpad for cursor control of touchscreen thumb operation in the mobile context. In: A. Marcus (ed.) Design, User Experience, and Usability: Users and Interactions, Springer International Publishing, Cham, pp. 563–574.

[51] Johnson E. (1965) Touch display–a novel input/output device for computers. Electronics Letters 1, pp. 219 – 220. DOI: `http://dx.doi.org/10.1049/el:19650200`.

[52] Enescu N.I., Mancas D. & Manole E. (2008) Locating gps coordinates on pda. pp. 470–474.

[53] Herodotou C. (2018) Mobile games and science learning: A comparative study of 4and 5 years old playing the game Angry Birds. British Journal of Educational Technology 49, pp. 6–16. DOI: `https://doi.org/10.1111/bjet.12546`.

[54] Loke L., Larssen A., Robertson T. & Edwards J. (2007) Understanding movement for interaction design: Frameworks and approaches. Personal and Ubiquitous Computing 11, pp. 691–701.

[55] American S. (2012), How has stephen hawking lived past 70 with als? URL: https://www.scientificamerican.com/article/stephen-hawking-als/. Accessed 25.4.2021.

[56] Al-Shathri A., Al-Wabil A. & Al-Ohali Y. (2013) Eye-controlled games for behavioral therapy of attention deficit disorders. In: C. Stephanidis (ed.) HCI International 2013 - Posters' Extended Abstracts, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 574–578.

[57] Smith J. & Graham T. (2006) Use of eye movements for video game control. p. 20.

[58] Apple (2020), Tracking face position and orientation. URL: https://developer.apple.com/documentation/arkit/arfaceanchor. Accessed 18.2.2020.

[59] Dorr M., Pomarjanschi L. & Barth E. (2009) Gaze beats mouse: A case study on a gaze-controlled breakout. PsychNology 7, pp. 197–211. URL: http://www.psychnology.org/429.php.

[60] Isokoski P., Joos M., Špakov O. & Martin B. (2009) Gaze controlled games. Universal Access in the Information Society 8, pp. 323–337.

[61] Damrongwatanapokin T. & Mikami K. (2018) Fatigue prediction and intervention for continuous play in video games. In: 2018 International Conference on Cyberworlds (CW), pp. 451–453.

[62] Juul J. (2011) Fear of failing ? the many meanings of difficulty in video games.

[63] Um S., Kim T. & Choi J. (2007) Dynamic difficulty controlling game system. IEEE Transactions on Consumer Electronics 53, pp. 812–818.

[64] Zohaib M. (2018) Dynamic difficulty adjustment (dda) in computer games: A review. Advances in Human-Computer Interaction 2018, pp. 1–12.

[65] Left 4 dead (series). URL: https://en.wikipedia.org/wiki/Left_4_Dead_(series). Accessed 11.5.2021.

[66] Csikszentmihalyi M. (1975) Beyond boredom and anxiety. Jossey-Bass, San Francisco, CA.

[67] Grand theft auto v. URL: https://www.rockstargames.com/V/. Accessed 11.5.2021.

[68] Hello Games (2018). No Man's Sky [Video game]. EN: Hello Games.

[69] Mojang (2009). Minecraft [Video game]. SE: Mojang.

[70] The State of Gaming App Marketing April 2020. URL: https://www.appsflyer.com/state-of-gaming-q1-2020. Accessed 31.7.2020.

[71] Yang Z., Li Z., Song L., Wu Q. & Fu S. (2014) Evaluation research of joystick in flight deck based on accuracy and muscle fatigue. In: D. Harris (ed.) Engineering Psychology and Cognitive Ergonomics, Springer International Publishing, Cham, pp. 428–436.

[72] Nintendo (1984). Zapper [accessory for the Nintendo Entertainment System]. JP: Nintendo.

[73] Baabdullah A, Bokhary D, Kabli Y, Saggaf O, Daiwali M, Hamdi A (2020) The association between smartphone addiction and thumb/wrist pain: A cross-sectional study. Doi: 10.1097/MD.0000000000019124.

[74] Bartle R. (1996) Hearts, clubs, diamonds, spades: Players who suit muds .

[75] Drachen A., Mirza-Babaei P. & Nacke L. (2018) Games User Research. Oxford University Press.

[76] Yun C., Shastri D., Pavlidis I. & Deng Z. (2009) O' game, can you feel my frustration?: Improving user's gaming experience via stresscam. pp. 2195–2204.

[77] Nosu K., Kurokawa T., Horita H., Ohhazama Y. & Takeda H. (2007) Real time emotion-diagnosis of video game players from their facial expressions and its applications to voice feed-backing to game players. vol. 4, pp. 2208 – 2212.

[78] Chukong Technologies, Cocos2d-x. URL: `github.com/cocos2d/cocos2d-x`.

[79] Taito (1986). Arkanoid [Video game]. JP: Taito.

[80] Erin Catto, Box2d. URL: `https://github.com/erincatto/Box2D`.

[81] Gregory J. (2018) Game engine architecture. crc Press.

[82] Allen J.J. (2005) Micro electro mechanical system design. CRC Press.

[83] Apple Inc. (2017). ARKit [Application Programming Interface]. US: Apple Inc.

[84] Rotberg, Ed (1978). Super Breakout [Video game]. US: Rotberg, Ed.

[85] Stefania Medryk I.S.M. (2013) A Comparison of Accelerometer and Touch-based Input for Mobile Gaming. Proceedings of the International Conference on Multimedia and Human Computer Interaction , pp. 117.1–117.8Corpus ID: 12646957.

[86] Natapov D. & MacKenzie I. (2010) The trackball controller: improving the analog stick. In: Future Play.

[87] Rajanen M. & Tapani J. (2018) A survey of game usability practices in north american game companies.

[88] Lewis J.R. (1995) Ibm computer usability satisfaction questionnaires: Psychometric evaluation and instructions for use. International Journal of Human–Computer Interaction 7, pp. 57–78. URL: https://doi.org/10.1080/10447319509526110.

[89] Barnett L., Gatzidis C. & Harvey C. (2017) An investigation into usability and first time user experiences within a mobile gaming context .

[90] Chalmers R., Begley C. & Caffery B. (2010) Validation of the 5-item dry eye questionnaire (deq-5): Discrimination across self-assessed severity and aqueous tear deficient dry eye diagnoses. Contact lens & anterior eye : the journal of the British Contact Lens Association 33, pp. 55–60.

[91] game genre V. (2020), Video game genre — Wikipedia, the free encyclopedia. URL: https://en.wikipedia.org/wiki/Video_game_genre, [Online; accessed 12-December-2020].

[92] Tirkkonen-Condit S. (1990) A think-aloud protocol study. In: Learning, Keeping and Using Language: Selected papers from the Eighth World Congress of Applied Linguistics, Sydney, 16 21 August 1987, vol. 2, John Benjamins Publishing, vol. 2, p. 381.

[93] Financial labs oy. URL: https://finlabs.fi. Accessed 3.5.2021.

[94] Hussain Z. & Griffiths M.D. (2009) The attitudes, feelings, and experiences of online gamers: A qualitative analysis. CyberPsychology & Behavior 12, pp. 747–753.

[95] Braun V. & Clarke V. (2006) Using thematic analysis in psychology. Qualitative research in psychology 3, p. 77–101.

[96] Atari, The atari mindlink. URL: http://www.atarimuseum.com/videogames/consoles/2600/mindlink.html. Accessed 15.4.2020.

9. APPENDICES

Appendix 1 Detailed results from the statistical analysis of the game measurements results

Table 11. Level completion results

Level	Interaction	Total played	Fail	Success	Success %
Easy	Touch	30	7	23	77
	Rotate	30	22	8	73
	Eye Closing	33	14	19	58
	Head Tracking	34	11	23	68
	Eye Tracking	60	45	15	25
Medium	Touch	36	21	15	42
	Rotate	21	7	14	67
	Eye Closing	36	25	11	31
	Head Tracking	34	24	10	29
	Eye Tracking	50	48	2	4
Hard	Touch	45	30	15	33
	Rotate	25	12	13	52
	Eye Closing	80	74	6	8
	Head Tracking	31	25	6	19
	Eye Tracking	53	53	0	0

Table 12. Level scores

Level	Interaction	Count	Sum	Average	Variance	p-value
Easy	Touch	23	171.78	7.47	0.16	$p = 0.623$
	Rotate	22	163.63	7.44	0.01	
	Eye Closing	19	140.77	7.41	0.01	
	Head Tracking	23	173.30	7.53	0.15	
	Eye Tracking	15	111.57	7.44	0.01	
Medium	Touch	15	198.97	13.26	2.30	$p = 0.481$
	Rotate	14	178.61	12.76	0.97	
	Eye Closing	11	142.28	12.93	1.18	
	Head Tracking	10	125.43	12.54	0.72	
	Eye Tracking	2	24.25	12.12	0.00	
Hard	Touch	15	293.06	19.54	5.09	$p = 0.082$
	Rotate	13	241.40	18.57	7.01	
	Eye Closing	6	109.60	18.27	4.49	
	Head Tracking	6	127.80	21.30	2.30	
	Eye Tracking	0	0	0	0	

Table 13. Successfully finished levels times

Level	Interaction	Count	Sum	Average	Variance	p-value
Easy	Touch	23	150.22	6.53	0.16	$p = 0.623$
	Rotate	22	144.37	6.56	0.01	
	Eye Closing	19	125.23	6.59	0.01	
	Head Tracking	23	148.70	6.47	0.15	
	Eye Tracking	15	98.43	6.56	0.01	
Medium	Touch	15	200.03	13.34	1.44	$p = 0.222$
	Rotate	14	194.39	13.89	0.29	
	Eye Closing	11	151.72	13.79	0.62	
	Head Tracking	10	140.57	14.06	0.02	
	Eye Tracking	2	27.75	13.88	0.00	
Hard	Touch	15	307.94	20.53	2.95	$p = 0.382$
	Rotate	13	274.60	21.12	0.59	
	Eye Closing	6	126.40	21.07	0.63	
	Head Tracking	6	121.20	20.20	1.02	
	Eye Tracking	0	0	0	0	

Table 14. Clockwise angles per second

Level	Interaction	Count	Sum	Average	Variance	p-value
Easy	Touch	22	3670.28	166.83	146.21	$p < 0.05$
	Rotate	22	765.04	34.77	189.08	
	Eye Closing	13	1418.36	109.10	341.92	
	Head Tracking	23	1244.44	54.11	45.06	
	Eye Tracking	15	568.55	37.90	2096.06	
Medium	Touch	15	2143.07	142.87	346.25	$p < 0.05$
	Rotate	14	381.39	27.24	265.63	
	Eye Closing	8	704.50	88.06	1564.59	
	Head Tracking	10	523.88	52.39	552.15	
	Eye Tracking	2	91.76	45.88	1358.08	
Hard	Touch	13	1339.78	103.06	985.55	$p < 0.05$
	Rotate	13	311.15	23.93	89.80	
	Eye Closing	1	78.59	78.59	0	
	Head Tracking	6	222.50	37.08	593.56	
	Eye Tracking	0	0	0	0	

Table 15. Counterclockwise angles per second

Level	Interaction	Count	Sum	Average	Variance	p-value
Easy	Touch	23	3675.76	159.82	359.98	$p < 0.05$
	Rotate	22	807.94	36.72	210.97	
	Eye Closing	19	2217.35	116.70	701.15	
	Head Tracking	23	1547.06	67.26	1586.94	
	Eye Tracking	15	957.24	63.82	1379.64	
Medium	Touch	14	1896.78	135.48	1001.17	$p < 0.05$
	Rotate	14	354.97	25.36	156.85	
	Eye Closing	11	1019.14	92.65	192.38	
	Head Tracking	10	458.65	45.87	450.26	
	Eye Tracking	2	115.06	57.53	168.59	
Hard	Touch	15	1647.63	109.84	1365.95	$p < 0.05$
	Rotate	13	330.66	25.44	157.18	
	Eye Closing	6	473.84	78.97	407.17	
	Head Tracking	6	231.50	38.58	361.63	
	Eye Tracking	0	0	0	0	

Table 16. Clockwise angles per turn

Level	Interaction	Count	Sum	Average	Variance	p-value
Easy	Touch	22	1924.67	87.49	1582.59	$p < 0.05$
	Rotate	22	334.52	15.21	61.46	
	Eye Closing	13	549.73	42.29	1061.41	
	Head Tracking	23	616.47	26.80	246.63	
	Eye Tracking	15	3.92	0.26	0.08	
Medium	Touch	15	947.55	63.17	709.83	$p < 0.05$
	Rotate	14	193.97	13.85	133.80	
	Eye Closing	11	340.60	30.96	2157.14	
	Head Tracking	10	227.07	22.71	89.63	
	Eye Tracking	2	112.75	56.38	3349.21	
Hard	Touch	15	462.09	30.81	264.15	$p < 0.05$
	Rotate	13	171.76	13.21	64.60	
	Eye Closing	1	1.10	1.10	0	
	Head Tracking	6	86.51	14.42	115.21	
	Eye Tracking	0	0	0	0	

Table 17. Counterclockwise angles per turn

Level	Interaction	Count	Sum	Average	Variance	p-value
Easy	Touch	23	1930.57	83.94	863.67	$p < 0.05$
	Rotate	19	270.76	14.25	26.82	
	Eye Closing	19	1196.11	62.95	1084.59	
	Head Tracking	15	399.13	26.61	412.84	
	Eye Tracking	15	548.95	36.60	474.25	
Medium	Touch	14	893.75	63.84	733.93	$p < 0.05$
	Rotate	14	153.78	10.98	34.52	
	Eye Closing	11	410.31	37.30	190.91	
	Head Tracking	10	198.21	19.82	68.88	
	Eye Tracking	2	84.61	42.31	4.71	
Hard	Touch	15	423.36	28.22	162.67	$p < 0.05$
	Rotate	13	148.75	11.44	30.65	
	Eye Closing	6	190.94	31.82	322.35	
	Head Tracking	6	101.90	16.98	108.12	
	Eye Tracking	0	0	0	0	

Table 18. Filtered movement count

Level	Interaction	Count	Sum	Average	Variance	p-value
Easy	Touch	30	3458	115	2930	$p < 0.05$
	Rotate	30	7681	256	2322	
	Head Tracking	34	6644	195	2678	
	Eye Tracking	60	11196	187	3452	
Medium	Touch	36	6260	174	12103	$p < 0.05$
	Rotate	21	12415	591	14410	
	Head Tracking	34	14030	413	25331	
	Eye Tracking	50	14367	287	13378	
Hard	Touch	45	4855	108	3498	$p < 0.05$
	Rotate	25	22292	892	34801	
	Head Tracking	31	16387	529	51939	
	Eye Tracking	53	21211	400	39494	

Table 19. Arcs left when level failed

Level	Interaction	Count	Sum	Average	Variance	p-value
Easy	Touch	7	8	1.14	0.14	$p = 0.129$
	Rotate	8	13	1.63	0.27	
	Eye Closing	14	22	1.57	0.57	
	Head Tracking	11	19	1.73	0.22	
	Eye Tracking	45	79	1.76	0.33	
Medium	Touch	21	66	3.14	3.13	$p < 0.05$
	Rotate	7	13	1.86	2.14	
	Eye Closing	25	67	2.68	2.64	
	Head Tracking	24	68	2.83	3.45	
	Eye Tracking	48	197	4.10	2.18	
Hard	Touch	30	126	4.20	7.41	$p < 0.05$
	Rotate	12	35	2.92	4.81	
	Eye Closing	74	316	4.27	6.94	
	Head Tracking	25	126	5.04	6.04	
	Eye Tracking	53	329	6.21	5.63	

Table 20. Minimum and maximum arcs left when level failed

Level	Interaction	Min left	Max left
Easy	Touch	1	2
	Rotate	1	2
	Eye Closing	1	3
	Head Tracking	1	2
	Eye Tracking	1	3
Medium	Touch	1	7
	Rotate	1	5
	Eye Closing	1	6
	Head Tracking	1	7
	Eye Tracking	1	7
Hard	Touch	1	9
	Rotate	1	7
	Eye Closing	1	10
	Head Tracking	1	9
	Eye Tracking	1	10

Appendix 2 Detailed results from the statistical analysis of the postquestionnaire results

9.1. Question: Overall, I Am Satisfied with How Easy It Is to Play This Game

Table 21. Question 1 answers

Level	Interaction	Average	Variance	p-value
Easy	Toucah	5.53	2.01	$p < 0.05$
	Rotate	6.00	1.50	
	Eye Closing	5.59	3.26	
	Head Tracking	4.82	3.03	
	Eye Tracking	3.12	1.49	
Medium	Touch	5.24	1.82	$p < 0.05$
	Rotate	5.88	1.36	
	Eye Closing	5.65	2.49	
	Head Tracking	5.41	1.88	
	Eye Tracking	2.82	2.03	
Hard	Touch	5.65	0.74	$p < 0.05$
	Rotate	5.65	1.74	
	Eye Closing	4.82	1.90	
	Head Tracking	4.59	3.01	
	Eye Tracking	3.24	1.69	

9.2. Question: It Was Simple to Play This Game

Table 22. Question 2 answers

Level	Interaction	Average	Variance	p-value
Easy	Touch	6.00	1.00	$p < 0.05$
	Rotate	5.94	1.18	
	Eye Closing	5.41	3.13	
	Head Tracking	4.94	2.81	
	Eye Tracking	2.94	1.93	
Medium	Touch	5.24	2.19	$p < 0.05$
	Rotate	5.94	0.93	
	Eye Closing	5.35	1.62	
	Head Tracking	5.24	2.44	
	Eye Tracking	3.12	2.24	
Hard	Touch	5.35	1.87	$p < 0.05$
	Rotate	5.65	1.99	
	Eye Closing	4.76	2.57	
	Head Tracking	4.76	2.44	
	Eye Tracking	3.18	2.03	

9.3. Question: I Could Effectively Complete the Objectives and Challenges

Table 23. Question 3 answers

Level	Interaction	Average	Variance	p-value
Easy	Touch	5.65	1.62	$p < 0.05$
	Rotate	5.94	1.68	
	Eye Closing	5.24	3.19	
	Head Tracking	5.12	2.24	
	Eye Tracking	3.18	2.03	
Medium	Touch	5.41	151	$p < 0.05$
	Rotate	5.24	2.19	
	Eye Closing	5.53	1.39	
	Head Tracking	4.65	3.24	
	Eye Tracking	2.94	1.68	
Hard	Touch	5.12	2.36	$p < 0.05$
	Rotate	5.24	2.44	
	Eye Closing	4.24	2.19	
	Head Tracking	4.24	2.32	
	Eye Tracking	2.76	2.44	

9.4. Question: I Was Able to Complete Objectives and Challenges Quickly

Table 24. Question 4 answers

Level	Interaction	Average	Variance	p-value
Easy	Touch	5.35	1.87	$p < 0.05$
	Rotate	5.53	2.64	
	Eye Closing	5.00	2.88	
	Head Tracking	5.06	2.93	
	Eye Tracking	3.35	1.37	
Medium	Touch	5.06	2.81	$p < 0.05$
	Rotate	5.18	2.28	
	Eye Closing	5.00	2.63	
	Head Tracking	4.47	2.89	
	Eye Tracking	2.88	1.36	
Hard	Touch	4.94	2.18	$p < 0.05$
	Rotate	5.41	1.88	
	Eye Closing	4.18	2.90	
	Head Tracking	3.88	2.49	
	Eye Tracking	2.88	1.86	

9.5. Question: I Was Able to Efficiently Complete Objectives and Challenges

Table 25. Question 5 answers

Level	Interaction	Average	Variance	p-value
Easy	Touch	5.35	2.37	$p < 0.05$
	Rotate	5.35	2.62	
	Eye Closing	5.06	3.43	
	Head Tracking	4.82	2.90	
	Eye Tracking	3.24	1.44	
Medium	Touch	5.00	2.38	$p < 0.05$
	Rotate	5.00	2.63	
	Eye Closing	5.12	1.86	
	Head Tracking	4.47	2.89	
	Eye Tracking	2.88	1.36	
Hard	Touch	4.88	2.49	$p < 0.05$
	Rotate	5.35	2.49	
	Eye Closing	4.06	2.43	
	Head Tracking	4.12	3.24	
	Eye Tracking	2.65	1.49	

9.6. Question: I Felt Comfortable Using This System

Table 26. Question 6 answers

Level	Interaction	Average	Variance	p-value
Easy	Touch	5.24	2.44	$p < 0.05$
	Rotate	5.12	2.74	
	Eye Closing	4.82	3.53	
	Head Tracking	4.35	2.87	
	Eye Tracking	2.94	0.68	
Medium	Touch	5.00	2.63	$p < 0.05$
	Rotate	5.29	2.85	
	Eye Closing	4.76	2.69	
	Head Tracking	4.35	2.49	
	Eye Tracking	2.88	0.99	
Hard	Touch	5.29	2.35	$p < 0.05$
	Rotate	5.41	2.51	
	Eye Closing	4.53	2.51	
	Head Tracking	4.41	2.63	
	Eye Tracking	3.18	1.15	

9.7. Question: It Was Easy to Learn to Play This Game

Table 27. Question 7 answers

Level	Interaction	Average	Variance	p-value
Easy	Touch	5.88	1.74	$p < 0.05$
	Rotate	5.53	2.51	
	Eye Closing	5.06	3.43	
	Head Tracking	4.59	2.63	
	Eye Tracking	3.00	0.88	
Medium	Touch	5.24	2.07	$p < 0.05$
	Rotate	5.47	2.14	
	Eye Closing	4.76	2.57	
	Head Tracking	4.82	2.28	
	Eye Tracking	3.53	1.39	
Hard	Touch	5.24	2.07	$p < 0.05$
	Rotate	5.41	2.76	
	Eye Closing	5.06	2.68	
	Head Tracking	4.82	2.15	
	Eye Tracking	3.47	1.89	

9.8. Question: Whenever I Make a Mistake in the Game, I Recover Easily and Quickly

Table 28. Question 8 answers

Level	Interaction	Average	Variance	p-value
Easy	Touch	5.47	1.89	$p < 0.05$
	Rotate	5.41	2.38	
	Eye Closing	4.71	2.47	
	Head Tracking	4.35	2.49	
	Eye Tracking	2.94	0.93	
Medium	Touch	5.35	1.99	$p < 0.05$
	Rotate	5.12	2.61	
	Eye Closing	5.00	1.38	
	Head Tracking	4.18	2.28	
	Eye Tracking	2.82	0.90	
Hard	Touch	4.94	2.31	$p < 0.05$
	Rotate	4.82	2.15	
	Eye Closing	3.76	2.44	
	Head Tracking	3.94	3.68	
	Eye Tracking	2.71	1.72	

9.9. Question: The Organisation of Information on the Game Screens Is Clear

Table 29. Question 9 answers

Level	Interaction	Average	Variance	p-value
Easy	Touch	4.71	2.97	p = 0.421
	Rotate	4.53	2.01	
	Eye Closing	4.29	2.22	
	Head Tracking	4.24	2.19	
	Eye Tracking	3.76	1.57	
Medium	Touch	4.47	2.26	p = 0.289
	Rotate	4.59	2.26	
	Eye Closing	4.59	2.13	
	Head Tracking	4.24	2.44	
	Eye Tracking	3.65	1.37	
Hard	Touch	4.41	1.88	p = 0.484
	Rotate	4.71	1.97	
	Eye Closing	4.12	2.11	
	Head Tracking	4.35	2.24	
	Eye Tracking	3.88	1.24	

9.10. Question: The Interface of This Game Is Pleasant

Table 30. Question 10 answers

Level	Interaction	Average	Variance	p-value
Easy	Touch	4.29	1.97	p = 0.465
	Rotate	4.47	2.01	
	Eye Closing	4.47	2.64	
	Head Tracking	4.12	1.99	
	Eye Tracking	3.65	2.37	
Medium	Touch	4.18	1.78	p < 0.05
	Rotate	4.53	2.14	
	Eye Closing	4.29	2.35	
	Head Tracking	4.29	1.85	
	Eye Tracking	3.18	1.40	
Hard	Touch	4.59	2.13	p < 0.05
	Rotate	4.53	2.14	
	Eye Closing	4.47	1.89	
	Head Tracking	3.82	1.78	
	Eye Tracking	3.18	1.03	

9.11. Question: I like Using the Interface of This Game

Table 31. Question 11 answers

Level	Interaction	Average	Variance	p-value
Easy	Touch	4.59	2.63	p = 0.205
	Rotate	4.35	1.87	
	Eye Closing	4.18	2.28	
	Head Tracking	4.12	1.99	
	Eye Tracking	3.41	2.13	
Medium	Touch	4.76	1.82	p < 0.05
	Rotate	4.65	2.37	
	Eye Closing	4.41	2.26	
	Head Tracking	4.18	1.90	
	Eye Tracking	3.06	1.18	
Hard	Touch	4.53	2.01	p < 0.05
	Rotate	4.76	2.82	
	Eye Closing	4.35	2.62	
	Head Tracking	4.00	2.38	
	Eye Tracking	3.24	0.94	

9.12. Question: While Playing the Game My Eyes Felt Excessively Watery

Table 32. Question 12 answers

Level	Interaction	Average	Variance	p-value
Easy	Eye Closing	1.59	0.76	p = 1.000
	Eye Tracking	1.59	0.76	
Medium	Eye Closing	1.41	0.51	p = 0.064
	Eye Tracking	2.06	1.43	
Hard	Eye Closing	1.65	0.87	p = 0.854
	Eye Tracking	1.71	0.85	

9.13. Question: While Playing the Game My Eyes Felt Dry

Table 33. Question 13 answers

Level	Interaction	Average	Variance	p-value
Easy	Eye Closing	1.59	0.76	p = 0.514
	Eye Tracking	1.82	1.40	
Medium	Eye Closing	1.53	0.89	p = 0.066
	Eye Tracking	2.29	1.85	
Hard	Eye Closing	1.88	2.36	p = 0.506
	Eye Tracking	2.24	2.32	